NPS55-86-020

NAVAL POSTGRADUATE SCHOOL

Monterey, California

TECHNICAL

KINEMATICS/DETECTION MODEL FOR A NAVAL SEMINAR GAME

HAIM ELDOR

OCTOBER 1986

Approved for public release; distribution unlimited.

Prepared for:
Naval War College
Newport, RI 02841

FedDocs
D 208.14/2
NPS-55-86-020

FedDocs
D 208.14/2:
NPS-55-80-020

NAVAL POSTGRADUATE SCHOOL
Monterey, California

Rear Admiral R. C. Austin
Superintendent

D. A. Schrady
Provost

The work reported herein was supported by the Naval War College.

Reproduction of all or part of this report is authorized.

This report was prepared by:

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY CA 93943-5101

REPORT DOCUMENTATION PAGE

ORT SECURITY CLASSIFICATION CLASSIFIED	1b RESTRICTIVE MARKINGS
ECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
DECLASSIFICATION/DOWNGRADING SCHEDULE	
RFORMING ORGANIZATION REPORT NUMBER(S) NPS55-86-020	5 MONITORING ORGANIZATION REPORT NUMBER(S)
NAME OF PERFORMING ORGANIZATION val Postgraduate School	6b OFFICE SYMBOL (If applicable) Code 55
7a NAME OF MONITORING ORGANIZATION	
ADDRESS (City, State, and ZIP Code) nterey, CA 93943-5000	7b ADDRESS (City, State, and ZIP Code)
NAME OF FUNDING/SPONSORING ORGANIZATION aval War College	8b OFFICE SYMBOL (If applicable)
9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER N0012486WR35588	
ADDRESS (City, State, and ZIP Code) ewport, RI 02841	10 SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO PROJECT NO TASK NO WORK UNIT ACCESSION NO

TITLE (Include Security Classification)
INEMATICS/DETECTION MODEL FOR A NAVAL SEMINAR GAME (UNCLASSIFIED)

PERSONAL AUTHOR(S)
ldor, Haim

TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
echnical	FROM ________ TO ________	October 1986	114

SUPPLEMENTARY NOTATION

COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	seminar game, naval wargame, kinematic, detection opportunity, two-sided game

ABSTRACT (Continue on reverse if necessary and identify by block number)

A computer program is described which is intended to provide real-time support for umpires onducting two-sided seminar wargames. The program is designed to operate on a microcomputer.

The game is assumed to take place between players at the Battle Force or Fleet Commander evel. The commander controls the movements and operations of battle groups, and in some ases task forces, at the theatre level. The group in each case will be those of the highest ossible level that still maintains the unique detection capabilities of the individual group. he groups are represented either by a collection of point units (say a carrier escorted by everal other platforms), or by an area (circular or rectangular) patrolled and/or screened any type of platforms (submarines, aircraft, etc.). The whole group moves on its track as ne unit.

The model presented performs the following analyses:

a) plans the tracks for all groups according to the players specifications,

DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION Unclassified
NAME OF RESPONSIBLE INDIVIDUAL Prof. Haim Eldor	22b TELEPHONE (Include Area Code) (408)646-3127
22c OFFICE SYMBOL Code 55Eo	

19. ABSTRACT

b) controls the timing for status changes: courses, speeds, EMCON plans, etc., and

c) investigates possible detection opportunities among all groups by checking intersections among all relevant opposing "area groups" and/or ranges to the opponents detection systems.

The final decision as to the actual detection (the realization of the detection opportunitiy) is left to the control (the umpire), using a display of any combination of groups desired. Once a detection is achieved, the model exits (at the umpire's instruction) to an engagement model (not discussed in this report). After the engagement results are determined, a return is made to the kinematics/detection model.

Table of Contents

Page

1 Introduction 9

2 The Seminar Game - General Concept

3 Kinematics/Detection Model Concepts

3.1 General 12

3.2 Guiding Principles 14

4 Information Flow Diagrams 27

4.1 General Box Diagram 27

4.2 Box Introduction and Explanations 28

4.3 Subprograms: Mathematical Algorithms, Functions, Input/Output 37

List of Figures

Figure 1 General Logical Block Diagram for a Seminar Game 27

Figure 2 General Information Flow Diagram 28

List of Programs and Functions

Page

Program 1 Coordinate Conversion from the Game (X-Y) into (Lat,Long) Values 64
Program 2 Coordinate Conversion from the (Lat,Long) into the Game (X,Y) Values 64
Program 3 Determination of the 4 Rectangle Corners using its Sides a and b 65
Program 4 Determination of the Remaining 2 Corners of a Rectangle Given 2 Corners and a Point on the Opposite Side 66
Program 5 Calculation of Corners of an Extended Rectangle Due to Active Sensors 67
Program 6 Determination of the Extreme Latitude and Longitude Points in a Group 68
Program 7 Comparison of Extreme Points of Two Groups for Screening Purposes 70
Program 8 Calculation of the Parameters A,B,C of a Line Passing Through Two Points 71
Program 9 Finding Extreme Points on a Circle 72
Program 10 Rotation of a Point X around a Point C by α 73
Program 11 Track Plans 2 and 3 Based on Destinations, and Speeds or Time of Arrivals Respectively 74
Program 12 Calculation of Lat and Long of a Point Y on a Great Circle Track to Find the Course towards It 76
Program 13 One Step Rhumbline Advancement 78
Program 14 Checking for Possible Detection Opportunity between Two Circular Area Groups (say No. m and n) 79
Program 15 Checking for Possible Detection Opportunity between Two Rectangular Area Groups (say No. m and n) 80
Program 16 Finding an Intersection between Two Rectangles 85
Program 17 Finding Whether a Point P is inside a Given Rectangle 87

			Page
Program	18	Checking for Possible Detection Opportunity betweena Rectangular (say m) and a Circular (say n) Area Groups	93
Program	19	Finding the Intersection between a Rectangle and a Circle	94
Program	20	Checking for Possible Detection Opportunity betweena Circular (C) Area Group (say No. m) and a Battle Group (BG) (say No. n)	96
Program	21	Checking for Possible Detection Opportunity between a Rectangular (R) Area Group (say No. m) and a Battle Group (BG) (say No. n)	98
Program	22	Checking for Possible Detection Opportunity between Two Battle Groups (say No. m and n)	100
Program	23	Checking for Passive (DF) Directional Measurement Opportunity (towards an Emitting Radar or Sonar) between Two Groups (say No. m and n)	104
Function	24	Calculation of the Distance between Two Points (x_1,y_1) and (x_2,y_2)	107
Function	25	Calculation of a Great Circle Length D between Two Points 1 and 2 on a Sphere	107
Function	26	Calculation to Find cos(θ) between Two Vectors $\overline{a}$ and $\overline{b}$	108
Function	27	Calculation of the Distance d of a Point (x_0,y_0) from a Line Ax+By=C	108
Function	28	Calculation of a Rhumbline Course (Co) between Two Points	108
Function	29	Direction Angle Calculation (Azimuth) between Two Groups (say No. m and n)	109

Nomenclature, Symbols and Explanations

A B C - Coefficients (plane geometry) for a straight line represented by $Ax + By = C$

a - Length of a rectangular area group usually crossing the group track (NM)

BG - An index for a Battle Group

b - Width of a rectangular area group (NM)

C - An index to designate the Control, the [illegible].

also

- Course angle in great circle track calculations.

c - An index for the center of group (center of a circle or a rectangle or the major unit in a battle group)

CA - An index for a circular area group

CS - Course, measured as an angle from North (positive clockwise).

D - An index for Data base or Disc

also

- Distance between two points on a great circle (in angular min).

d - Distance between two points or from a point to a straight line

Det - A flag to indicate (when equal 1) a detection opportunity between two groups

DF - Short for Direction Finder

also,

- A flag designating a directional measurement opportunity against an active group or unit.

Dx - A distance to a point x on a great circle.

ESM - Passive detection equipment. Will be sometimes used for both radar and sonar emission detection.

i,j - Indexing a group by size (i=1, battle force, [illegible]) and a group within a battle force (j=1, ...)

INTER - A flag equal 1 or 0. A value of 1 indicates that an intersection between two areas was found.

- Short for Keyboard

[illegible] – [illegible] group [illegible]
[illegible]

Lat_1, Lat_2 – Latitudes of two points along a great circle for track planning.

Lat – Latitude (in ° or in angular min for each) according to need.

Lat, Long – The corner coordinates of a rectangular area [illegible] of the [illegible] in a single group.

$Lat_0, Long_0$ – A reference point in the center between the [illegible] display and the Lat, Long coordinate system.

$Lat_c, Long_c$ – Center coordinates.

$Lat_E, Long_E$ – Coordinates of the furthest east point of a group.

Lat_N – Latitude of the furthest north point of a group.

$Lat_R, Long_R$ – The corner coordinates of a radar centered rectangular area group.

Lat_S – Latitude of the furthest south point of a group.

$Lat_U, Long_U$ – The corner coordinates of a [illegible] rectangular area group.

[illegible] – Latitude and longitude of a point along a great circle.

$Lat_W, Long_W$ – Coordinates of the furthest west point of a group.

LO_1, LO_2 – Longitudes of two points along a great circle for track planning.

Long – Longitude (in ° or in angular min) according to need.

M – [illegible]

[illegible] – Maximum platform speed (kts).

[illegible] – Numbers assigned to pairs of [illegible] analyzed for detection opportunity. The [illegible] values stand for another numbering system for the groups in addition to the [illegible] system.

[illegible] – Status of radar or sonar [illegible] can assume the [illegible]

[illegible] – [illegible]

[illegible] – Stands for [illegible]

[illegible] – [illegible]

[illegible] – Circle radius [illegible]

[illegible]

– [illegible]

Rad - An index for radar.

Rec - An index for a rectangular area group.

R_p - Passive detection range of an emitting radar or sonar.

S - Short for sonar.

SCP - A flag indicating (when equal 1) that an active group is screened from further analysis for direction measurement opportunity by another group.

SCF - A flag indicating (when equal 1) that a group is screened from further analysis for detection opportunity by another group.

Son - An index for sonar.

Sp - Speed (kts).

T - Time (hrs).

T_e - Patrol end time (hrs).

TOA - Time of arrival at a destination.

T_r - Time for radar turn on or off (hrs).

TranSp - Transit or cruising speed (kts).

T_s - Time for sonar turn on or off (hrs).

T_{st} - Patrol start time (hrs).

T_{Along} - Time point along a great circle track.

Typeg - Type of the group arranged (battle group, rectangular or circular area groups).

T_{plat} - Type of platforms in the group (CVN, DD, Sub, P3, Air Field etc.).

U - Number of units in a battle group.

u - An index to designate a particular unit in a battle group.

V - Vertex point on a great circle.

[illegible] - A point on a great circle.

XY - Rectangular plane coordinates (n.mi).

(x_0, y_0) - Reference point for conversion between the great circle and rectangular coordinates.

x_{34}, y_{34} - A point on the line connecting points 3 and 4.

a_1, b_1, a_2, b_2 - Values defined for vectors (a_1, b_1) and (a_2, b_2) for calculation of the angle between the two vectors.

x_e, y_e - Corner points of an extended range rectangular area (due to radar or sonar operation).

[illegible] – [illegible]

[illegible] – [illegible]

[illegible] – Change in course [illegible]

[illegible] – Radar range and extension of a ground target line to radar operator [illegible]

[illegible] – Sonar range and extension of a ground target line to sonar operator [illegible]

[illegible] – Time to get from one point to another in [illegible] units

[illegible] – Time increments of track plotting

[illegible] – Calculated angle between two vectors

Kinematics/Detection Model for a Naval Seminar Game

1. Introduction

1.1 The model developed and presented is expected to be incorporated in a comprehensive seminar game.

1.2 The seminar game will take place between players at the level of battle forces and / or fleet commanders, controlling the movements and operations of battle groups, and in some cases, task forces at a theatre level.

1.3 The discussed model will keep track of the movements of aggregated groups at the highest possible level that still maintains the unique detection capabilities of the individual groups. A group can consist of quite a few units if they move essentially together. Alternatively, a group can be an individual unit or few units operating, say, as a patrol at a given area.

1.4 The model will not require tactical decisions of the players. In cases where some tactical considerations are involved, they are for the most part indirectly incorporated into the model structure itself. For example, a group searching for a target (or in a screening role using some tactics to cover a search area) is represented by the area covered without specifying exact unit locations. Tactical situations that will nevertheless arise should [illegible] treatment by the umpire.

1.5 The model will run with regular interrupts to allow both the players and the umpire to add or change decisions as necessary.

16. The processes supported by the discussed model are

(a) Movements (kinematics) of all blue and orange battle groups. (It is not recommended at this point of the model development to introduce neutral groups; however, such incorporation is not difficult and may be accomplished in the future.)

(b) EMCON plans.

(c) Detection opportunities between groups.

2. The Seminar Game - General Concept

2.1 The as model developed is a part of a seminar game series, which is schematically shown in the block diagram in Figure 1. A more detailed block diagram will be presented in Chapter 4.

2.2 The five main conceptual blocks in Figure 1 are:

(a) Separate seminar rooms for each decision making player (P) communicating only with the umpire (control).

(b) Game control (C) room (the umpire) with communication links to both the players and the computer. The control will load the computer with the relevant models and obtain the results. The control can overrule model recommendations and screen the players' decisions.

(c) The computer models (M), with access to all permanent auxiliary routines. The models will be able, at each step of the game, and by request of the umpire, to retrieve necessary data from the data base, concerning platforms, weapon systems, detection equipment etc. It can also activate output routines, both as a part of the game progress and for post-game analysis.

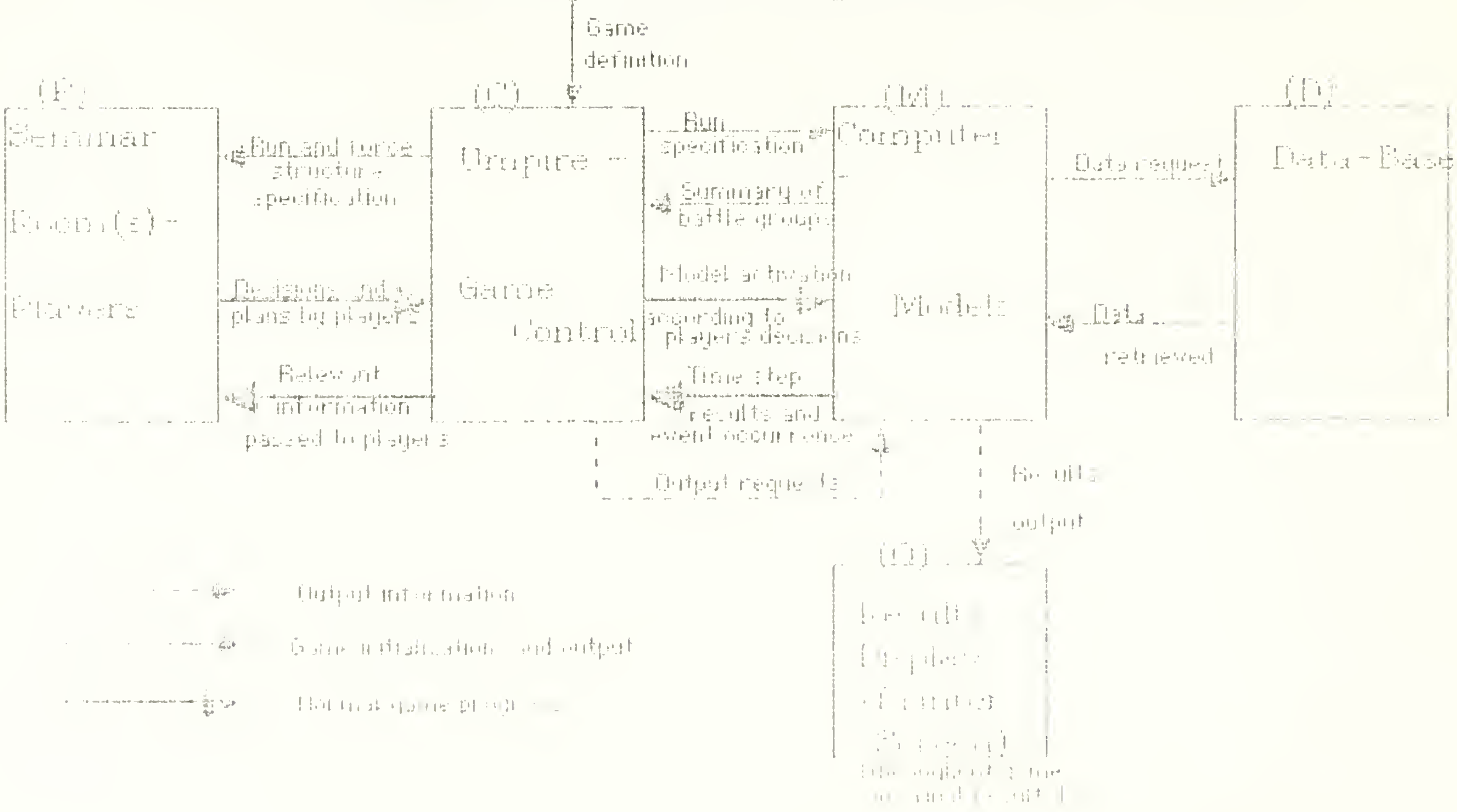
(d) The data-base (D), being accessible by the computer, will supply the model, in a predetermined data structure, all the data required for the run.

(e) The final results output (O) unit, activated by the umpire via the computer, will provide any output summary and desired display. Most of the output, at least in the kinematics/detection stage, is expected to be processed continuously as the game progresses. However, final summary and statistical results should be obtained with emphasis on the engagement results (not discussed in this report).

1.3 While a full characterization of the whole seminar game structure and routines is beyond the scope of this report, the modeling of any routine or model in the computer block (M) and in particular the kinematics/detection model) cannot be detached from the understanding of the interrelations among the five blocks. In particular, the general information flow, as will be discussed later is of a great importance in structuring the detection model.

Figure 1

General Logical Block Diagram for a Seminar game


Game definition
(P)
Seminar Room(s) - Players
(C)
Umpire - Game Control
(M)
Computer - Models
(D)
Data - Base
(O)
Results
Displays
Run and force structure specification
Decisions and plans by players
Relevant information passed to players
Run specification
Summary of battle groups
Model activation according to player's decisions
Time step results and event occurrence
Output requests
Data request
Data retrieved
Results output
Output information
Game initialization and output
Normal game progress

3. Kinematics/Detection Model Concepts

3.1 General

In the following pages we will summarize the concepts on which the model is based. First, in this chapter, a summary of the principles that guided us in constructing the model will be presented. Following, in the next chapter, we will introduce the information flow block diagram for the kinematics/detection model (Figure 2), separated into two parts: the part carried out by the computer model itself (below the separating broken horizontal line) and the part representing the people interacting with the computer, i.e., the umpire (the control) and the players. Each box is then described in the necessary detail, one by one, including a local block diagram whenever necessary, using programs, algorithms and functions listed thereafter in the next chapter. Note that the vertical broken line appearing in the diagram separates the game setup that may be prepared long before the actual game takes place (left of line) from the actual game progress (right of line).

3.2 Guiding Principles

3.2.1 Representation of Battle Groups

a) The definition of force groups as used throughout the report are as follows:
   - A *Player* or *side* represents the whole blue or orange opposing forces in the game. They are controlled by the actual players (say fleet commanders).
   - A *Battle Force* is a large group of units, usually of several types, operating under one command. We expect that in most cases the game will not include more than two battle forces for each side.
   - A *Group* of units in the the battle force has a particular task which is different from that of other groups in the same force. A group is the lowest level of elements represented in the

model is best when detection ranges for particular units are needed.

- The group will be represented either by a typical geometric representation of the area it covers, or by the unit positions in the group (more details will be presented in the next chapter).

b) The groups represented are those that have unique roles and have individual characteristics with respect to detection capabilities. Separation into individual units in the group is made only if it is needed for a detection process.

c) There are two types of groups discussed in the report:
- Battle groups represented by a collection of points (representing units) moving on parallel tracks.
- "Area groups" represented by areas (circles or rectangles) patrolled by some unit or units (whether surface, submarines or aircraft). The area group is represented by its geometric parameters rather by the units themselves, as no attention is given to the actual whereabouts of the units inside the area. Area groups will usually accompany a major battle group for screening purposes, moving, therefore, on a parallel track. However, independent area groups can just as well be incorporated in the game. In particular, mine fields may be represented as independent stationary area groups.

d. The area groups will usually move at the same speed as the battle groups they accompany, even though the units in the area group themselves may have different speeds (in some cases, such as for aircraft, a much higher speed).

e) We have ignored the transit step of fast units (say aircraft) from some base location to their assigned patrol area. We do this for the following reasons:

- The move is fast relative to the actual game progress.
- This step is relatively insignificant for the kinematic detection model described here.

- Attack aircraft should be considered as a part of the engagement model rather than of the detection model
- The transit stage of patrolling aircraft has more of a logistics significance than a distinct operational one
- Avoiding the incorporation of high speed units enables simplification of the model.

f) The model should be flexible in terms of changing group structures. In particular it is expected that a battle group may need to be split into say two sub-groups, each having a different task. The umpire should add the new groups cancelling the old one. Combining two groups into one should be similarly possible.

g) The decision as to how to treat a group structure when the course is changed, is not yet finalized. The two options are:
- Immediate rotation of the whole group (both a battle group or an area group) by the course change to maintain similar force structure relative to the new course. The algorithm for such a rotation is provided in this report.
- All units (or corners of a rectangular area) change direction to the new course retaining their old positions. Thus, a different orientation relative to the group course is obtained, but orientation relative to the threat axis is maintained.

(Of course, the change of a course of a circular area group is trivial).

No decision is made at this point as to the preference between the two approaches. Possibly they could be left to the game control or to the players, leaving both options in the model.

3.2.2 Establishing Detection Opportunities

a) Each group will be assigned a detection range. The detection range for an area group will be the defined as the basic area itself that it patrols.

b) A battle group moving under emission control (radars and active sonars "off") has no detection range.

c) Activation of radars or active sonars (to "on" position) provide a battle group with a detection range (around the emitting unit) according to the radar characteristics. For area groups it will extend the area (for detection opportunities) by an additional δ (say the radar range) beyond the original basic area shape which is based on passive sensors only. This will be defined as the extended area or a group with an extended range.

d) We will distinguish between detection (determination of target position) and direction measurement (determination of the bearing to the target).

e) A detection opportunity is established if any point or unit of a group is inside the detection range of some group of the opposite side. Detection opportunity between two opposing area groups is established if the two areas intersect each other. Note that detection can occur if the basic area (with unextended range) of one group intersects the extended detection range of the searching group. Intersection of two extended areas by only their extension regions does not provide detection opportunity to any of the two groups.

f) Detection opportunity does not necessarily imply actual detection. It is up to the umpire to judge whether the opportunity warrants detection. Note that the patrolling units in an area group cannot cover continuously all the area, so detection has a probabilistic nature. In general it is understood that a small intersection would have a low probability for an actual detection. A use of a large display of the tactical picture (e.g. showing the intersecting areas) will help the umpire in his judgement.

g) The above is also consistent with the treatment of mine fields represented by area "groups". In this case the "detection" opportunity stands for possible actual "hit". Note that the mine field area group does not have an extended range, nor can it be detected by all sensors.

h) The model detection procedures (sensors range definitions intersections, etc.) are unified for all possible configurations of opposing groups, whether surface, sub-surface, aircraft or mine fields. The detection opportunity identified by the model and presented to the umpire, should be accompanied by an indication of the types of the two groups involved and their mode of operation (active or passive). The eight group detection combinations are - surface/surface, surface/sub-surface(SS), surface/air, SS/SS, SS/air, air/air, mines/surface and mines/SS. The umpire will have to use that information to decide whether the opportunity really exists. (For example, a submarine not at periscope depth generally cannot detect an aircraft even when the areas intersect.). The umpire will have to use also additional considerations such as the types of ships involved, towed array deployment, ASW planes using sonobuoys, length and frequency of submarine periscope depth periods etc.

i) An important remark to be made here is, that two passive battle groups (with no radar or sonar operating, and ignoring communication emissions that are not included in the model), being represented each only by a collection of point units, and having therefore no detection range, may in principle cross each other (in the model) with no detection opportunity. This may not be a realistic situation, however:

- The probability for both battle groups to pass by their opponent's screening area groups without being detected is essentially zero.
- In most cases, at least one of the battle group units (perhaps a surveillance aircraft) will be active.

3.2.3 Simulation Time Step

a) It was initially thought that a variable time-step simulation would be suitable for this simulation. This should have solved the following problems:
- Long group transit periods when "nothing happens" could be achieved in one simulation step
- Busy periods with many "happenings" could be broken into several simulation steps

b) However, further considerations suggested the following advantages of a fixed time-step simulation approach:
- Easy control and kinematics management by the model
- More opportunities for the players and the umpire to change orders and decisions.
- Easier interaction with additional models (say the engagement model) in the future
- Simpler algorithms. There is no need to calculate the time of the next occuring event (for example the time when two tracks come within detection range).
- A sufficiently small time-step can be selected, so that it is essentially impossible for groups to cross without a detection opportunity

c) At the same time, the advantages of the event-controlled simulation do not form strong disadvantages for the fixed time-step one:
- Large time-steps on the display may be somewhat confusing. The umpire may want to be able to watch the advancement of the forces, say each real time hour, so it will be useful to have automatic program interrupts for the umpire's review. He may, at this point, just press a key for simulation continuation, but he has a chance to intervene (add, change, modify or take any other action).
- Many happenings in a short period involve mainly tactical decisions, which are not within the scope of this semi-strategic model.

d) It is estimated that one hour real time-steps will be adequate for the model. As long as no detection opportunity exists (when the forces are far from each other), the computation speed is expected to be high, and the umpire will be able to advance the simulation quite fast if he so desires.

e) Also, a one hour step can move the groups by not more than roughly 20-30 NM, a short distance relative to the areas covered by all groups, so there is little chance for detection opportunity loss due to two groups crossing each other between time-steps.

3.2.4 Track Calculations (Kinematics)

a) The one hour time-step simulation enables us to perform each time-step force advancement using a local regular plane geometry representation with no significant error.

b) However, when determining the transit track between points more than several hundred miles apart, great circle calculations may be required. This calculation should only be performed once for each track plan, with the resulting intermediate positions and courses stored for future local plane calculations.

c) The model shall provide the players with various options for track plans, such as -

- given a destination and a desired TOA calculate speed and/or course changes

or

- calculate the arrival time at a particular point given course(s) and speed(s)

etc.

The plans are described in detail in the next chapter in Box 1.

d) EMCON plans should be specified in a similar manner: changes are specified at given time-steps (turn on, turn off, turn on momentarily for a radar 'glimpse').

e) Note that all plans should be specified by the players in terms of units of time-steps, even though in events could occur at any time. The one hour steps are expected to be sufficient for the level of game considered here.

3.2.5 Data-Base

a) The data-base should include all data concerning the platforms that may be included in the seminar game - surface, sub-surface or aircraft.

b) The list of the necessary parameters for the kinematics/detection model is given elsewhere (see description of box D next chapter).

c) It is important to construct a special data-base for the seminar game consistent with the model data requirements. A particular recommended format is not presented here, as it will be strongly dependent on the actual program structure.

d) It would also be desirable to have theatre maps stored for display during the game.

e) Either in the data-base itself or on some additional storage device, one can keep previous game setups for display or reuse.

3.2.6. Graphical Displays and Associated Calculations

a) It would be a realistic approach to have the players specify all their plans in terms of geographical Lat. and Long. coordinates in the same manner that it is done at sea.

b) However, many of the model calculations are easily performed in a regular X-Y coordinates with an agreed upon local origin.

c) It is essential that (at least) the umpire will have the capability to observe on his monitor, at least a partial display of the force structure, whereabouts, the area groups and in the future perhaps, also the theatre map. It is therefore necessary to have the proper conversions from the Lat/Long and/or X-Y coordinates into the display and back. Such conversion will also enable the umpire to input geographical data using a mouse. Details are not fully described, in the report being strongly dependent on the computing system used and the mapping features that may be eventually incorporated.

4 Information Flow Diagram

4.1 General Box Diagram

In this chapter we will present the general kinematics (detailed) or block diagram (Figure 2). For most of the model blocks, we have supplied "local" flow diagram. Note though, that the flow diagrams can aid the understanding of the model structure, but caution should be taken in using them directly for programming even though in some cases they are presented at what seems to be a programming level.

Programs of lower level than those appearing in this chapter, as well as mathematical algorithms and functions are described separately in section 4.3.

In Figure 2 note:

a) The region above the horizontal broken line represents the information processed outside the computer by the umpire and the players.

b) The region under the horizontal broken line represents the information processed by the model itself

(The arrows crossing the horizontal line are therefore the interactions between the umpire and the computer).

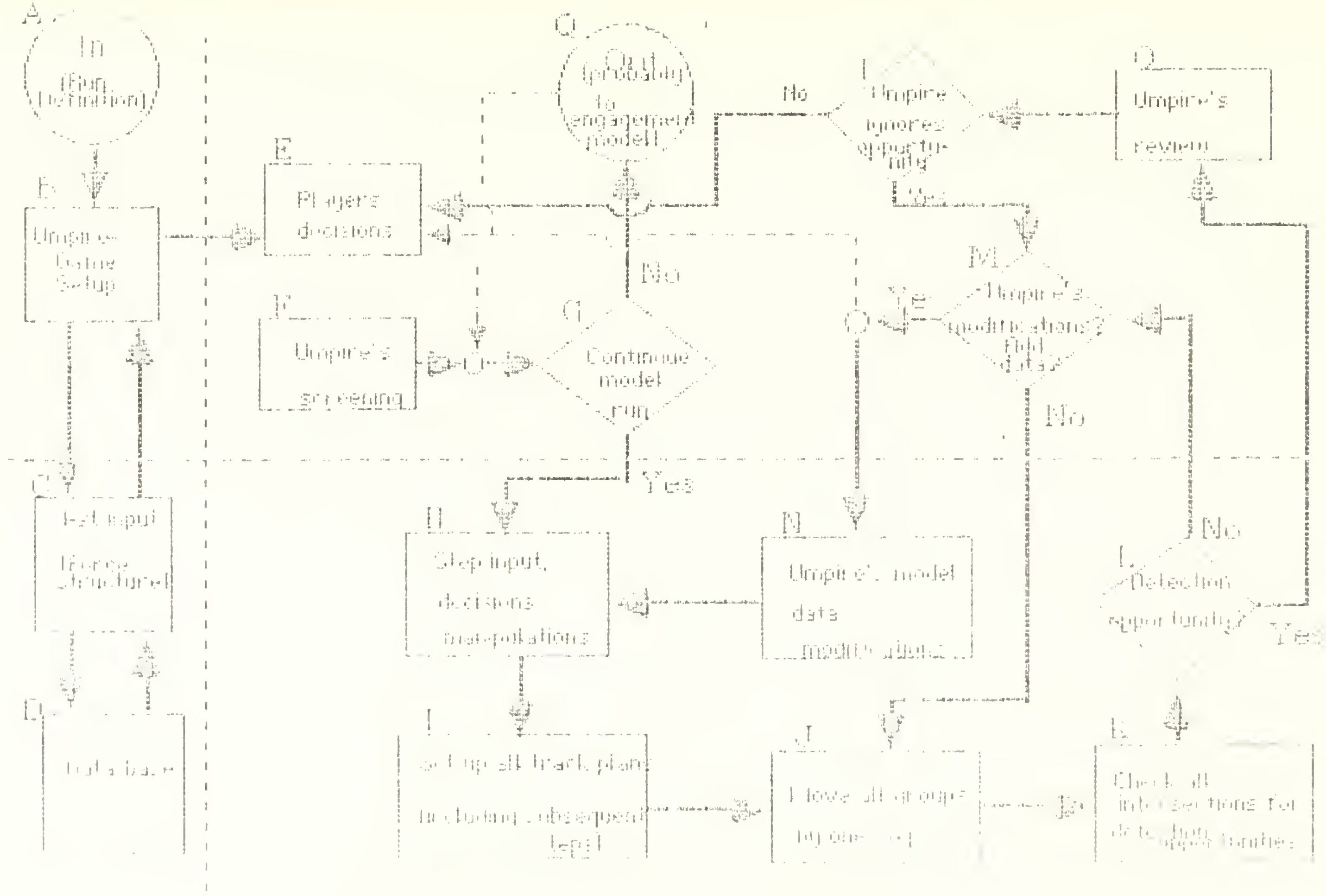
c) Left of the vertical broken line we can find the procedures that can be separated from the model run, and can be therefore prepared long before the actual game takes place

d) Right of the vertical broken line - the actual game progress

Remark. Throughout this report quite often "variables" are expressed in some computer coding notation; also, quite a few block diagrams are presented at least partially, in a seemingly programming format. However, it is not the intention of the writer to dictate any particular coding approach. The notation and the block diagrams are used for convenience of illustrating some principles for the program structure. It is understood that in the coding stage itself, significant improvements can (and should) be made above the approach outlined here, bearing in mind the need to save computing time, and in particular storage space. Furthermore, in many cases the present description is incomplete for coding needs; for example, the representation of all state variables, with all the indices, of the various battle groups, carrying each group all associated information and data is fully left for the programming stage. The manner in which they are treated in this report is merely for demonstrating the general concept.

Figure 2

General Information Flow Diagram



4.2 Box Introduction and Explanations

(Refer to Figure 2)

4.2.1 Box A - "In" Step

a) This box represents the information delivered to the umpire possibly by higher hierarchy as to:

- The general purpose of the game.
- The participating fleets and their general structure.
- Support forces availability to each side.
- Initial position of all forces in general terms.
- Fleet mission orders.

b) All the above input is not in any computer compatible format but in a regular military order form.

c) Note that the game may be initiated also by the control group itself (the umpire).

4.2.2 Box B - Game Setup (by the umpire)

a) All information obtained for game initialization is arranged in tables representing the force grouping.

b) It is suggested that groups are (roughly) defined using an i,j,k indexing as follows:

i=1,2 - to denote blue or orange side

j=1...5(?) - to denote up to 5 large forces for each side (in most cases - probably not more than 2)

k=1...10(?) to denote up to 10 battle groups (or area groups) within each force

Note that the final grouping will be done by the players decisions in box E.

c) Altogether we may expect the game to include up to 2×5×10=100 groups. It is anticipated that in most cases there will be not more than about 2×2×8 = 32 active groups.

d) The umpire will designate to each group its type (say TYPEP(i,j,k)) through a code value standing for CVN, DD, SSN, P3, mine field etc. Note that in some cases a battle group may consist of more than one type of units, even though they may not be all needed as separate units in the kinematic/detection model. However, the detailed specification of their types may be useful for the engagement model not described in this report. Also individual units that may be eventually operating say a radar or a sonar within the battle group, should be specified.

e) Later, an additional code will be needed for each group (say TYPEG(i,j,k)) to designate the group arrangement type as a battle group, rectangular or circular area group. This will be done though only following the players decisions as to their force grouping.

f) Once all groups and types are defined, the umpire will activate the 'data fetching' routine. box D (fetching the required data from the data-base)

4.2.3 Box C - Data Fetching Routine

a) The data fetching routine is activated by the umpire and is supplied with a list of TYPEP(i,j,k) codes by which it will return from the data-base the following data for all groups according to the units consisting the group:

- Cruise/transit speed.
- Maximal speed.
- Type(s) of radars
- Types of sonars.
- Types of passive sensors (ESM against radars or passive acoustic)
- Typical flight altitudes (for planes !?)

b) For each detection system type (radar, sonar, ESM, etc.) range tables are arranged to include ranges against all possible opposing unit types. Note that the tables can be arranged in terms of the units carrying the radars (say) instead of the radars themselves. Visual range tables may also be [illegible] arranged at this point.

c) We foresee the need for five types of tables:

- Ranges of all active sensors (radars, sonars) of side 1 against all possible unit types of side 2
- The same for side 2 against side 1
- Ranges of all passive sensors (ESM, passive acoustic) of side 1 against all active radars or sonars of side 2
- The same for side 2 against side 1
- Visual ranges of all units against all opposing units

Whenever a detection range will be required in the model it will be selected from these tables according to the pair of unit types involved and their mode of operation (active or passive).

d) All parameters fetched from the data base will be identified as (say):
- TRANSP(i,j,k)
- MAXSP(i,j,k)
- RAD(i,j,k) (=Radar code number)
- SON(i,j,k) (=Sonar code number)
- ESM(i,j,k) (=ESM sensor code number)

etc

Note that the radar, sonar etc code numbers can be used to approach the range tables.

e) Additional data that will be eventually needed for other models such as the engagement model may include:
- Missile types
- Guns
- Modes of firing whenever relevant
- Hit probabilities(?), at least as an aid to the umpire in his decisions.

Also, data for logistics calculations will be eventually needed, such as:
- Stocks (missiles, ammunition)
- Reloading capabilities and times to carry out
- System reliabilities(?)

etc

All these are neither described nor discussed in this report.

f) The data-base may contain ready data for map display, for say several typical theatres or of a previously run game. It is expected though that map drawing will be added to the program only at a later stage.
Note that map display may be useful not only towards a visual understanding of the game progress by the umpire, but also towards the possibility of group (or unit) location input using a mouse on the map. We do not elaborate on the mapping display features but we will provide here the

mathematical algorithms needed for the plot of data input on the screen.

g) Once all data - force structure (including unit data where as necessary), the range tables, and theatre information (map) is ready, it can all be saved on a disc for a future run.

h) The model should also supply at this point (by the umpire's request), a full printout list of the whole force structure, as well as two separate lists for the two players (passed to them by the umpire). For the umpire, possibly a regular display on the screen may be satisfactory (instead of a printout).

4.2.4 Box D - Data-Base

a) The data base should include all information that may be required for the game about all units - surface, sub-surface and aircraft.

b) The data should be arranged in a predetermined format compatible with the model input requirements.

c) A possible arrangement is e.g., in strings of data as follows: code number (to designate the type of unit), cruising speed, max speed, type(s) of radar, ESM, sonar, types of weapon(s) with firing rates, ammunition quantities (if relevant), susceptibility to damages, fuel consumption rate, visual detectability (ranges with few typical conditions - day, night, fog), etc.

d) For the kinematics/detection model only part of the above information is needed:

- Type of unit (CVN,DD,........) in code .
- Cruising speed.
- Maximal speed.
- Type of radar(s).
- Type of sonar.
- Type of ESM sensor.
- Visual detectability (possibly at two or three levels of visibility).

e) For each radar type a full list of nominal detection ranges against all relevant possible opposing units must be included in the data-base.

f) The same must be done for all detection systems. For ESM sensors the table should be based on possible opposing radars rather than units.

g) Data for map drawings for prespecified games.

4.2.5 Box E - Players Decisions

a) At this box decisions are made by each player with respect to:

- Change of locations of forces or groups, including the possibility of redefinition of the groups (this can be done only on the first step of the simulation - the initial deployment step).
- Track plans for all groups for any future time length desired. This should be done in simulation time-step units. Plans can be changed throughout the game progress whenever the players are approached by the umpire.
- EMCON plans which will include specified times (in simulation time-step units) for radar or sonar turn 'on' or 'off' or 'glimpse' mode of operation. Again, prespecified plans can be changed during the game. If no plan is supplied, the model will use the default option of all sensors 'OFF' (silent mode) till otherwise specified.
- A decision following an indication from the umpire about detection of an opponent group or unit (whether actual detection or just a direction measured towards an emitting, say opponent's radar). Such decision could be for example splitting a group as a preparation for engagement or for deception purposes.

b) The decisions passed to the umpire could be in the form:

Track

- Battle force No. ..., group No. ... change course to ... or change speed to ......

or

- Battle force No. ..., group No. ... new destination lat. [illegible] to arrive at (date, hour), or [illegible] towards it in speed [illegible]

EMCON

- Battle force No. ..., group No. ... turn 'on' radar type ... (or turn 'off' or 'glimpse') at particular hour ...

Group rearrangements

- Battle force No. ....., group No. ....., units ..... and ..... to split from main group with course ..... and speed ..... ; a group change course to .... (and/or speed to ....).

c) Decisions concerning direct engagement are not summarized here.

d) The decisions can be made any time the umpire approaches the players. It would usually be when some kind of an 'event' has occurred, triggering some changes in the environment.

e) However it would be possible to have the players see the game clock (say in one hour units) and let them initialize a change in decisions by letting the umpire know about it ahead of time. Obviously, the umpire himself can, using his judgement, allow the players make any change or modification at any time (representing say an order change, or a new intelligence information etc.). This feature will be introduced again upon discussion of Box M.

f) It should be emphasized again that all model interrupts, and therefore all changes/additions/modifications can be made only at the model 1 hour time steps.

4.2.6 Box F - Umpire's Screening

a) All players' decisions must pass the umpire's screening to detect:

- Moves not in accordance with the game "rules" (e.g. travel plans stated in a manner not processable by the model; or usage of time specifications not in the 1 hour step format)
- moves that cannot be incorporated in the game (e.g. moves outside the theatre "map")
- Errors with respect to support forces availability etc.

Upon detection of such errors the umpire will inform the player about the need to modify his instructions.

b) The umpire must also screen out player's instructions that are irrelevant to the model. Such instructions will be ignored by the umpire during the model input process (Following Box [illegible]

c) The umpire will coordinate moves by the two sides so as to make the game meaningful (e.g. when it is obvious to him that no engagement, or maybe not even detection by either side is anticipated by their instructions)

4.2.7 Box G - Continuation Decision and Input to the Model

a) The kinematics/detection model run continuation decision by the umpire has meaning at some later stage of the game, in particular when:

- Detection (by either side) caused activities needing a different micromodel deployment (say the engagement model). Exit into Box Q is initiated at this point. Note though that the engagement may involve only part of the forces so a return to Box G from Box Q (to the kinematics/detection model) should be possible, to the same state at which it was left. All game state variables should be therefore stored at this point for possible game continuation from the same state, except for some of the variables that will have to be changed due to the engagement results.
- The umpire may willfully decide to stop the game (at least temporarily) for analysis.
- The umpire feels the game should be restarted due to some errors made (improper definition of the game scenario changes or maybe just input keypunching errors).

b) In general, the umpire will at this point input the data needed for the next time-step calculations to be carried out in the model. These will be summarized in the next box (Box H).

c) Activating the first step (the initialization of the game) will require an extensive data input. Subsequent steps however will need input for just the changes made (if at all). In principle though, all inputs can be entered at each step (say every game one hour), except for moving the forces from one location to another. Placement of forces can (and must) be made only at the initialization step. An exception to this rule is the input of location for a new group whose participation is to take place only at a later stage of the game, and can be entered therefore in the midst of the game progress.

d) The types of inputs that may be entered by the [illegible] box are as follows:

Geographic locations of forces (both sides)

For each group (i,j,k) - player, battle force No. and group No. - enter the location - (Lat, Long) at time 0, either through the keyboard or with aid of a mouse on the map.
Note: the (i,j,k) indices should already carry the types of units in the group, with all the needed data about sensors, weapon systems etc. as listed in Box 5.
The location (Lat, Long) will be assigned to a central geometric point of the group - the center of the circle or the rectangle, or the position of the say, the [illegible] of the group.

Group configuration

For each group (i,j,k) define the configuration type by say -
BG - a battle group,
Rec - a rectangular area group,
Cir - a circular area group.

A battle group should be entered as a list of the locations (Lat,Long) of each of the units in the group, preceded [illegible] by the number defining the group size, i.e. the number of units in the group (in addition to the central unit). The data can be entered through the keyboard or by the mouse (directly on the display unit) provided the map feature will be operational. The final representation will be:
(i,j,k),BG,n,Lat_0,$Long_0$,(Lat_m,$Long_m$),$m=1$, to n

A rectangular area group can be defined in two ways:
(i) The center (already entered), lengths of the two rectangle sides, a and b ("a" being say the long side), and the orientation (in degrees) of the rectangle relative to the group course (clockwise). Such input will be entered through the keyboard and look as follows:

i,j,k,Rec,a,b,α

(ii) Using the mouse for input, one can enter two points defining one of the rectangle sides (say "a"), and any point on the opposite parallel side. The rectangle is thus defined. The input will therefore be:

$i,j,k,Rec,Lat_1,Long_1,Lat_2,Long_2,Lat_x,Long_x$

where x is any point on the opposite side (the 3-4 points line).

Note: the center of the rectangular area group (previously entered) will have now to be modified according to the new rectangle corners .

The final group representation in the model will be (after the required calculations described later in Box H are made):

$i,j,k,Rec,Lat_0,Long_0,(Lat_m,Long_m)_{m=1\ to\ 4}$

A circular area group can be also entered in two ways:

(i) By the keyboard - enter radius of circle to previously entered center ($Lat_0,Long_0$). Input:

i,j,k,Cir,R

(ii) By the mouse, specify any point x on the circle

$i,j,k,Cir,Lat_x,Long_x$

The point x will be used to calculate (see Box H) the circle radius. .

The final circular area group representation in the model will be:

$i,j,k,Cir,Lat_0,Long_0,R$

Group extended range configuration

This configuration will need the ER and SR (active radar and sonar ranges). However, no additional input by the umpire is needed as these values are directly retrieved from the range tables previously arranged with the battle force definition.

Track planning (according to players instructions)

For each group (i,j,k) use one of the following track plans input format:

Track plan (1) - a sequence of pairs of values - speed (Sp) and course (Co), with time specifications (selected on the time-step 1 hour units, not necessarily each hour). Example

$i,j,k,1,T_0,Sp_0,Co_0,T_1,Sp_1,Co_1,\ldots$

↑

plan

Track plan (2) - sailing towards given destinations ($Lat_m,Long_m$) in a given speed. The model enables the user to supply one or several destinations, and it will calculate changes in course for five time-steps (accommodating a great circle sail by rhumbline tracks changing every 5 hours). An input example:

$i,j,k,2,Lat_1,Long_1,Sp_1,Lat_2,Long_2,Sp_2,\ldots$

↑

plan

Track plan (3) - sailing towards given destinations (final or intermediate ones), arriving at given times - TOA. Again course change will be made every 5 hours. An example:

$i,j,k,3,Lat_1,Long_1,TOA_1,Lat_2,Long_2,TOA_2,\ldots$

↑

plan

Track plan (4) - use the same course and speed for group (i,j,k) as for group say (l,m) previously calculated. This may be mainly used for a group (i,j,k) serving as a screening group for a main battle group (l,m) of the same side. It can, however, also be used for tracking an opponent's group after detection. Examples:

i,j,k,4,l,m (a screening group) or

i,j,k,4,2,l,m (side 1 group tracking side 2 group)

↑

plan

Track plan (5) - use data supplied for another group of the same side, when both groups must sail to the same destination (probably at the same time). Example:

i,j,k,5,l,m

↑

plan

Notes

1) To each plan, when it is applied to an area group, values of Tst and Te may be added to designate start time (Tst) and end time (Te) of the patrol.

2) A plan for a mine field area group is needed only with the T_{st} value, specifying its Sp to be equal to 0, i.e.

i,j,k,1,T_{st},0,0

EMCON plans

All active emitting equipment will be assumed to be in an "off" position till otherwise requested by the player(s). All passive equipment is assumed to be continuously "on" with no need for such specification in the model. Radar and sonar activation control plan can be made using one of the following modes:

Modes 0-1 - turn the radar (or sonar) "off" or "on" respectively for continuous silence or operation till a change time is reached in the simulation. The times for changes (in simulation time step units) must be specified (for each group separately) as say, Tr's and Ts's as shown in the next examples:

i,j,k,Radar,Tr,1,Tr,0,Tr,1,......

or similarly

i,j,k,Sonar,Ts,1,Ts,0,Ts,1,......

Mode 2 provides the option for occasional (on the hour) glimpse. Example:

i,j,k,Radar,Tr,2

This mode can be used only when the equipment is in an "off" position.

Combination instructions can be conceived in the form, say,

i,j,k,Radar or Sonar, Tr,2 Tr,1 Ts,1 Tr,0, Tr,1,Ts,0, ...etc.

Whenever a radar or a sonar is turned on (mode 1 or 2), detection ranges are available for the proper needed calculations according to the pair of groups analyzed for possible detection opportunity.

e) All inputs can be rearranged by [illegible], etc. [illegible] alternatively, one can combine all data pertaining to a given [illegible] — Location, configuration, trail and [illegible] different structure, according to convenience and [illegible] efficiency in the coding process.

4.2.8 Box H - Input Processing by the Model

a) The input by the umpire (as described in the preceding Box G) may contain:

- The geographic location and group configuration (Block H1) is usually done only at the first game initialization; occasional addition of groups is allowed also at later stages.
- Track planning for each group. The plans can be changed by the players throughout the game progress. A general outline is shown in Block H2.
- EMCON plans - can be also changed at will. See a general outline in Block H3.

Notes:

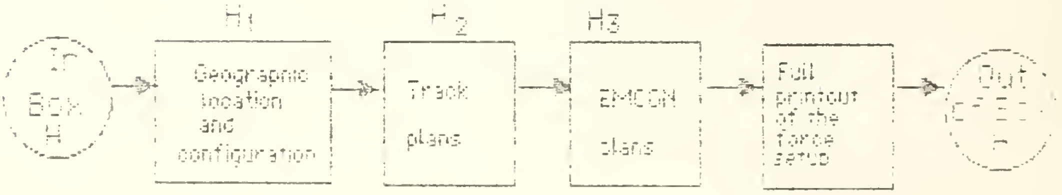
1) Whenever a change is made, all previous calculations related to that change are nullified. In particular, tracks may have to be recalculated.

2) Note that we have not indicated the option of bypassing this stage just as it is not done for any piece of input when it is not needed or not changed. It is reasonable to assume that the preferred model input structure will be based on input menus presented to the umpire.

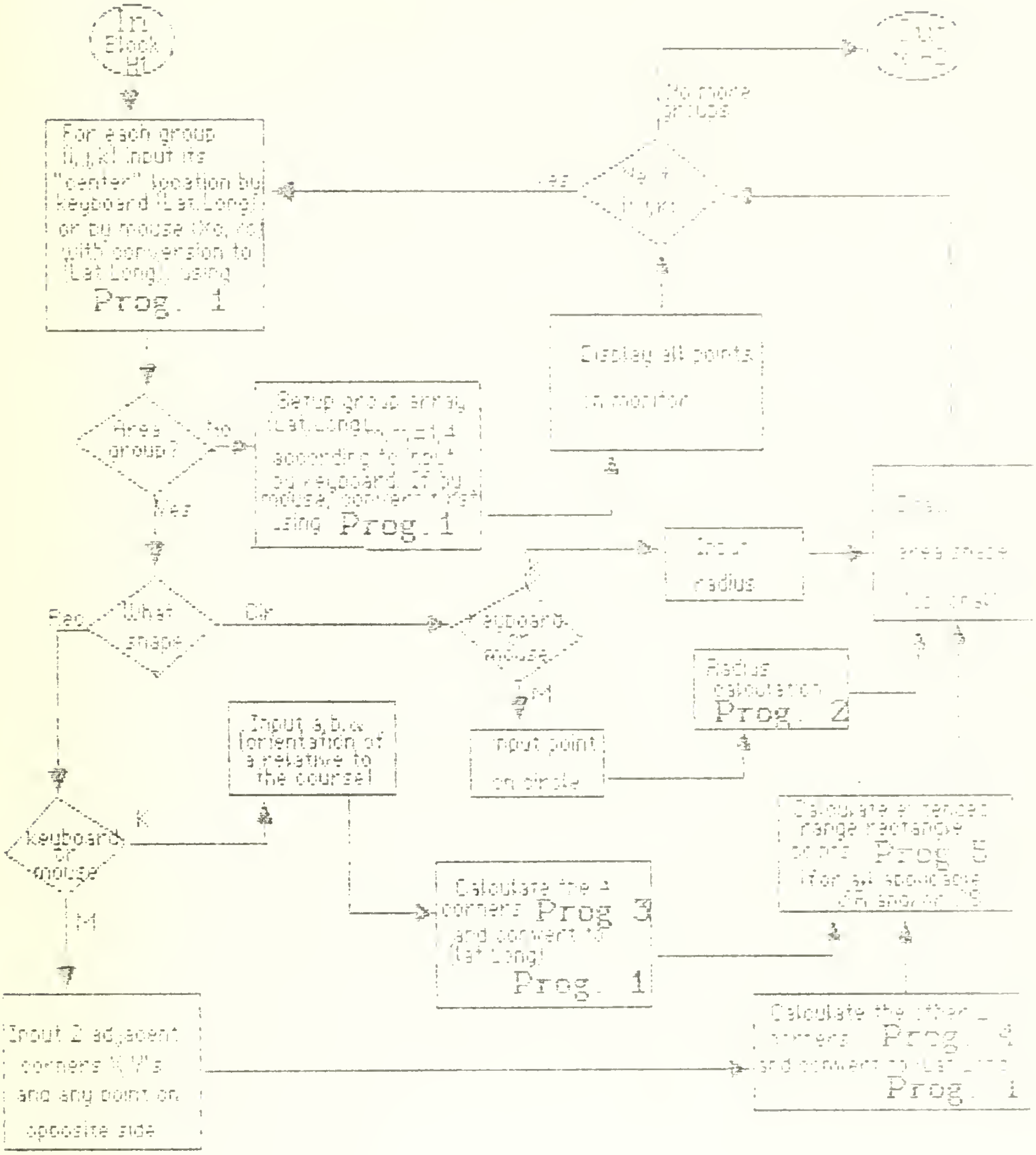
3) Since this box is passed when some changes are made, an optional force configuration and (new) setup printout (or display to the umpire) should be possible at this point.

The general structure of this box (Box H) is presented here. Each sub-block, H1, h2, and H3 is described in the diagrams in the next pages.

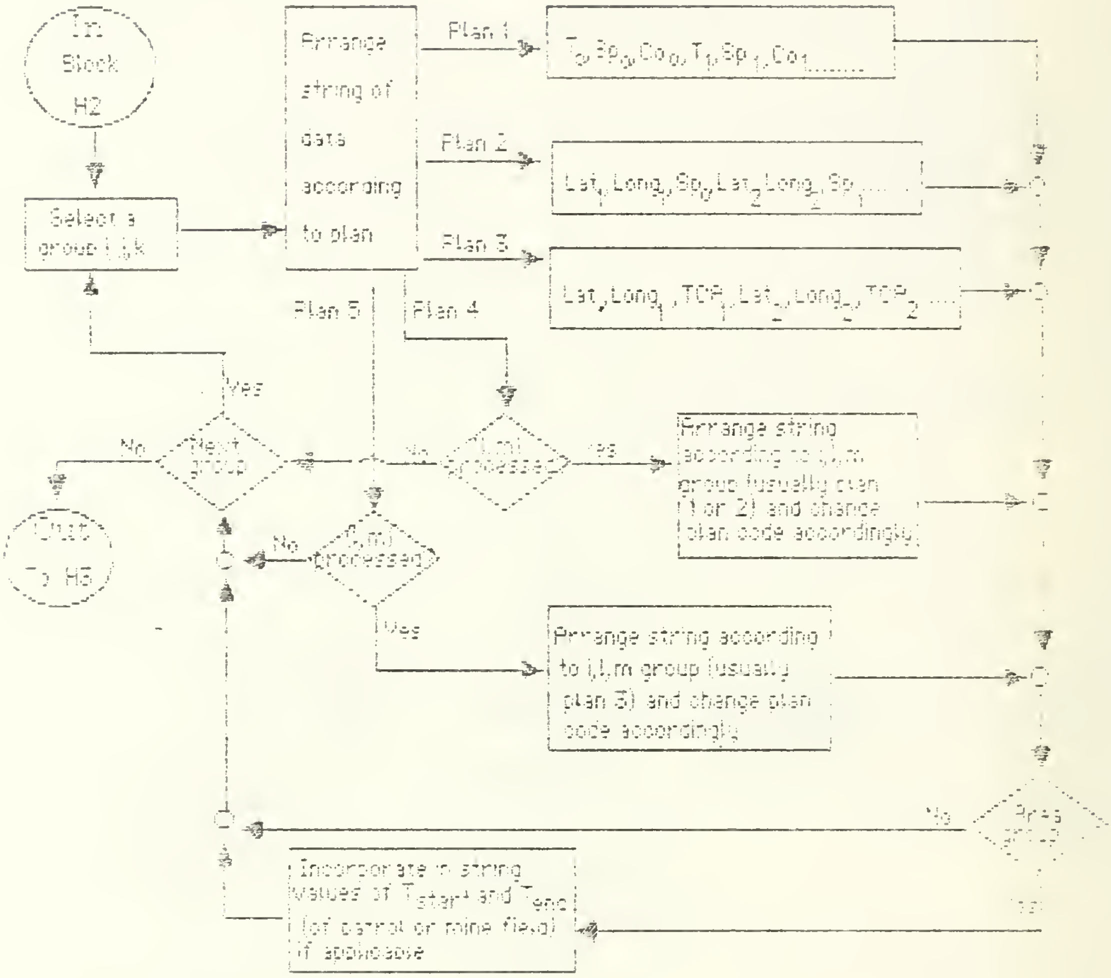
b) The general block diagram for Box H is as follows:


H1
H2
H3
In Box H
Geographic location and configuration
Track plans
EMCON plans
Full printout of the force setup
Out of Box H

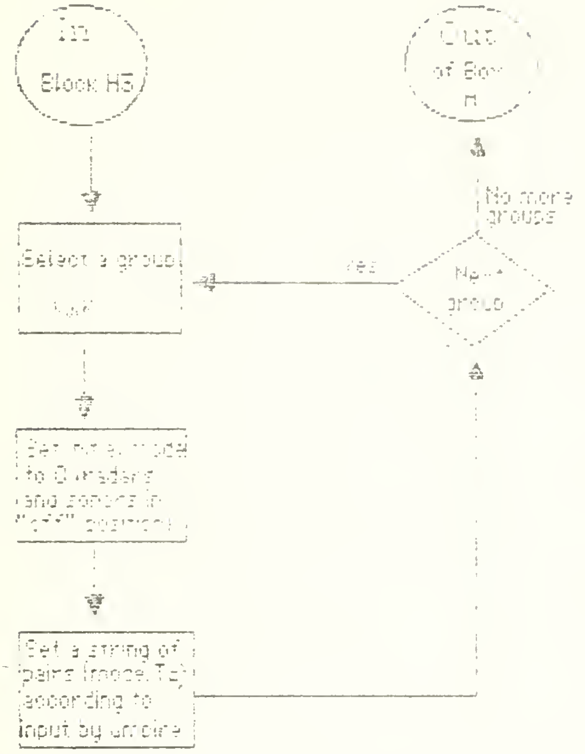
c) Sub-block H1


In Block H1
For each group (i,j,k) input its "center" location by keyboard (Lat,Long) or by mouse (Xo, Yo) with conversion to (Lat,Long) using Prog. 1
Area group?
No
Setup group array (Lat,Long), i=1,4 according to input by keyboard. If by mouse, convert first using Prog. 1
Yes
What shape?
Rec
Cir
keyboard or mouse
K
M
Input radius
Input point on circle
Radius calculation Prog. 2
area inside circle
keyboard or mouse
K
M
Input a, b, α (orientation of a relative to the course)
Calculate the 4 corners Prog 3 and convert to (Lat,Long) Prog. 1
Input 2 adjacent corners X,Y's and any point on opposite side
Calculate the other 2 corners Prog. 4 and convert to (Lat,Long) Prog. 1
Calculate extended range rectangle points Prog 5 (for all boundaries on shoreline)
Display all points in monitor
Yes
No more groups
Out

d. Subblock H2


In Block H2
Select a group i,j,k
Arrange string of data according to plan
Plan 1
$T_0, Sp_0, Co_0, T_1, Sp_1, Co_1, \ldots$
Plan 2
$Lat_1, Long_1, Sp_0, Lat_2, Long_2, Sp_1, \ldots$
Plan 3
$Lat_1, Long_1, TCP_1, Lat_2, Long_2, TCP_2, \ldots$
Plan 5
Plan 4
Yes
No
Next group
No
(l,m) processed
Yes
Arrange string according to i,l,m group (usually plan 1 or 2) and change plan code accordingly
Exit To H3
No
(l,m) processed
Yes
Arrange string according to i,l,m group (usually plan 3) and change plan code accordingly
No
Yes
Incorporate in string values of T_{start} and T_{end} (of patrol or mine field) if applicable

e) Sub-block H5



Note. The data treated in this sub-block is almost automatically arranged in the proper string through the umpire's input.

4.2.9 Box I - Track Planning

a) For each group, the requested track calculations will be made according to the track plan code (1-3) as indicated in Box H.

b) Codes 4-6 are changed to to one of the codes 1-3 according to the players' instructions

c) The output of block I should consist of a sequence of times (on the hour times but not necessarily each hour) with two additional parameters - new speed and course at each time T_i. This is done for each group to yield:

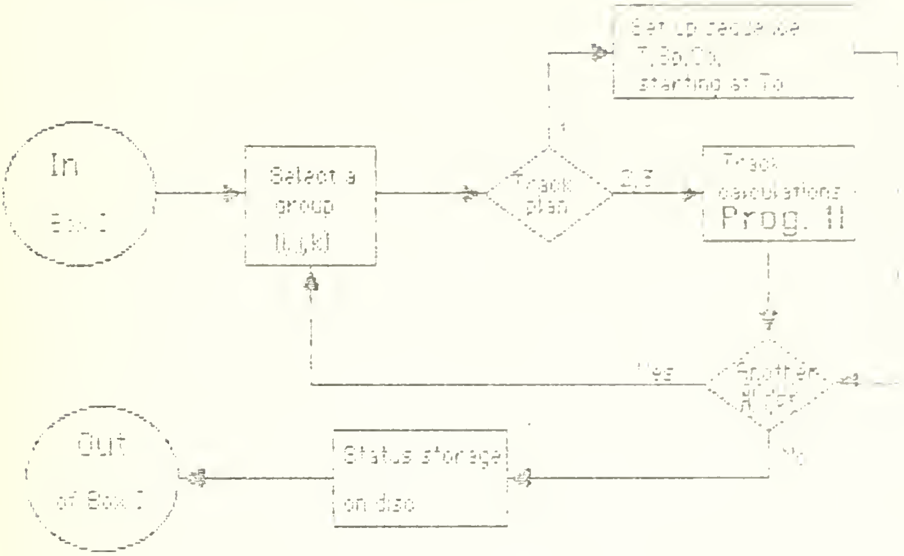
$i,j,k,T_0,Sp_0,Co_0,T_1,Sp_1,Co_1,........$

Note: The input values (in track plans 2 and 3) of Lat_i and $Long_i$ of the destinations, are not considered any more. They will be eventually recalculated according to the actual track achieved by the sequence of speeds and courses (see Box J).

d) Box I can be executed as a part of Box H, for each group one by one. It is presented here seperately for the sake of clarity. It is also believed that separation is preferred for computational and coding simplification.

e) The results of boxes H and I yield state conditions - speed, course, detection equipment position, to start move calculations at each simulation step (say every one hour). they should be preserved for all subsequent steps for which the apply, the first one always erased with each move. Whenever a change in the overall plan is made, the whole sequence of data is erased to be replaced by a new one

f) Block diagram for Box I



4.2.10 BOX.U - Game Track Progress

a) Each move for each group, is made according to the presently known values of Sp and Co for the given time T (read from the data string T_1,Sp_1,Co_1; T_2,Sp_2,Co_2, etc...). At each point in time (in simulation time-steps), the values of Sp and Co may be updated (if necessary).

b) The moves are made one step at a time (one hour steps, to the next position. The one hour step calculations are based on simple local plane geometry with rhumbline progress. Note that there could be therefore small deviations in the step final values of the Lat and Long as compared to the desired ones. however, this well represents what really happens at sea. Also, the deviations are negligible in terms of the game scope.

c) In battle groups the actual units will be moved directly by translating their old positions (say Lat_{k-1},$Long_{k-1}$) into a new position (Lat_k,$Long_k$).

d) If a change of course is made for any group the model allows two possible procedures:

- To maintain the same group structure relative to the course, a rotation of unit positions around the center unit can be made by an angle equal to the course change. One must remember that the model will carry out this rotation at "no time".
- The whole group will move now in the new course without any rotation, thus creating a new group structure relative to its course (which may be a more realistic representation).

e) Area groups will move as follows:

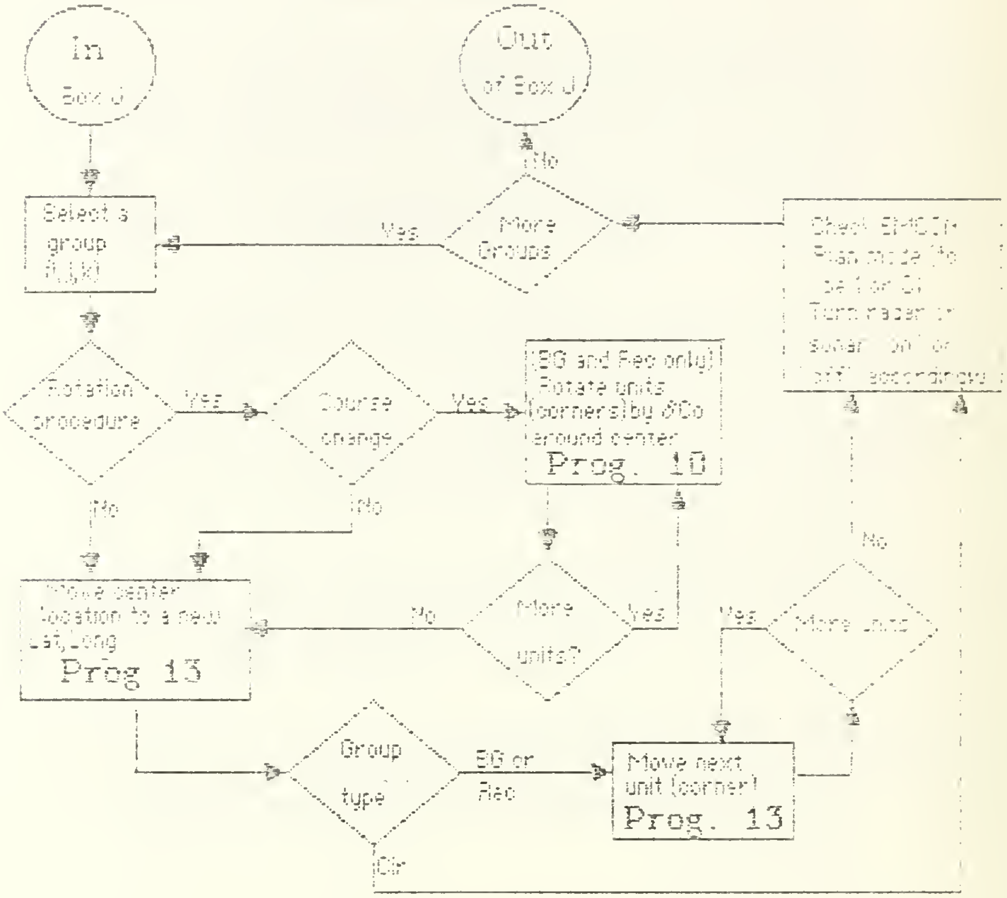
- For a rectangular area, move the center and the four corners to the next location (rhumbline) according to the course and speed. If needed and desired, a rotation procedure by (11) can be made in the same manner described earlier. The rotation should be made prior to the move. Note that a rotation in the case of an area group is more realistic since it does not represent actual physical locations of the units but rather a patrol area that in principle can be in most cases rotated at "no time" (with the exception may be of a submarine or a boat that can be left out of the area after the rotation...)

- For a circular area group, just move the center point to the new location. Rotation has no meaning in this case.

f) We do not provide any strong argument for the way to handle the change in course procedure. The decisions that must be made are:

- Will the model be fixed with one of the two procedures (with or without rotation), and which one.
- Will it be the umpire's or the players' decision as to what procedure to use.
- Will the procedure be unified for all groups or left flexible to be used differently for each group say by the players' instruction.

g) Block diagram for Box J



4.2.11 Box X - Detection Opportunity Analysis

a) In this box checks are made between all pairs of groups of opposing sides in order to identify detection (and counterdetection) opportunities. Such opportunity, when it is identified, does not automatically mean actual detection. It will be the umpire's prerogative to decide whether the opportunity is realized.

b) We will differentiate between two types of possible detections:

- Detection of a passive target by passive means (such as passive accoustics) or active means (radar, sonar).
- Establishment of direction to an active target (emitting radar or sonar signals) by passive means such as ESM and passive acoustic respectively.

Detection of passive targets

c) Since the targets in the model can be of two types - a battle group (a collection of units) or an area group, two types of procedures must be supplied for the passive target case:

- A battle group may be detected if at least one of its units is found inside the detection range of the searcher.
- An opportunity to detect an area group target is identified if there is an intersection between the target passive area configuration (the potential area for physical presence of the target) and the detection range area of the searcher. Note that the detection range can be either the basic area configuration when only passive detection measures are used, or an extended area due to an increased range when active radars and/or sonars are used.

d) Some of the detection opportunities depend on operations, tactics and/or force structure. For example

- In its mission, how often is the submarine expected to be at periscope depth, exposing itself to (mainly) radar detection, but also capable of radar and ESM operation?
- Does any of the surface platforms in the group carry a towed array?
- Do the planes in an air area group have ASW capacity or are they just visual search and surveillance planes?

The above information should be specified as a part of the game definition to provide the umpire with tools for his decisions with respect to the detection opportunity realization. Note that tactics are not incorporated in the model, so the above type of information need not in general be specified as input to the model (system detection ranges will be automatically entered anyhow).

e) Due to the screening process that is explained below (at item g)), most of the times an intersection check is made there will be a high probability for the existence of detection opportunity. When such an opportunity is analyzed, the smallest search area (i.e. the intersection between the two basic areas) will be first analyzed, to eliminate the need for further analysis if detection opportunity was already identified at this range. If the opportunity was not identified, the model should now check the next larger detection range (up to the longest active range). The group configurations (including the extended ranges) and the intersection display will aid the umpire in his decision as to the actual detection.

f) Note that the intersection check between pairs of groups should be made twice - with search area of each one against the basic physical area of the other, except for the check for intersection of the two basic areas that must not be repeated twice and supplies detection opportunity to both sides.

g) In the model, there will be no need to check area intersections between all opposing groups. A first screening [illegible] should be made to check distances by [illegible] on [illegible] basis of extreme points of the analyzed groups (e.g. [illegible] the furthest north of one with the furthest south of the [illegible]). In such a way, exclusion of cases where intersections cannot occur, is made.

Detection of Active Targets

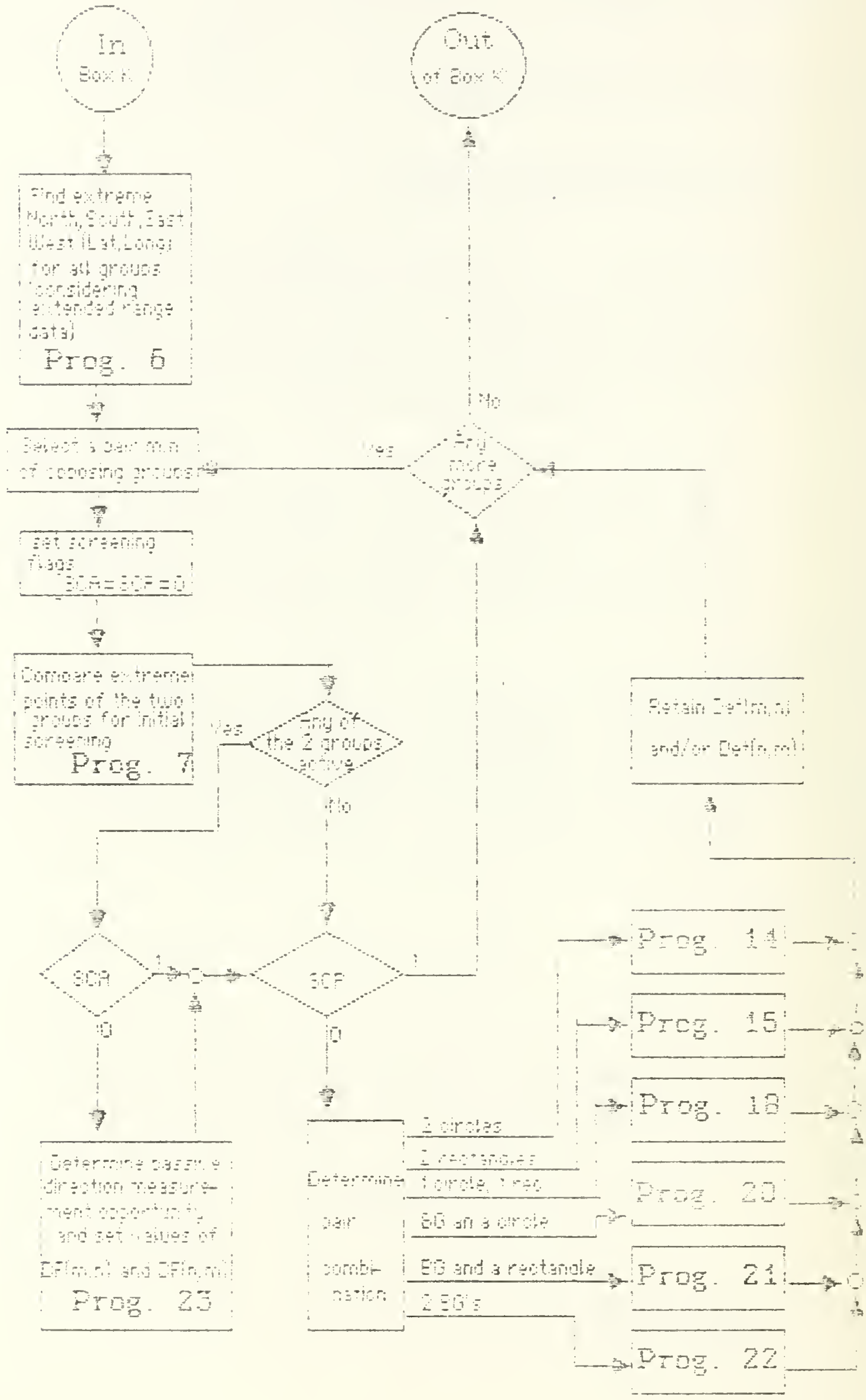
4) In addition to the passive target possible detection by the intersection analysis, an active target can be sensed by [illegible] equipment (against an operating radar) or passive acoustic means (against an active sonar). The detection provides only directional measurement (however several directions may be used to locate the target by a triangulation procedure; the triangulation procedure is not done by the model, and it will be the umpire's decision as to the actual detection. Possibly the players themselves will be able to perform the calculations for the triangulations.

5) Directional information is obtained at longer ranges. Therefore, regardless of the area intersections, a range check should be always made between the emitting group and the closest point of the searching group, to identify an opportunity for a passive detection of an active target.

6) In the case of an active area group, the most extreme point (e.g. one of the closest rectangle corners) will be [illegible] its distance to the closest point of the searching group. In case of an active battle group, where in practice only one unit is really operating its radar, only this particular unit (and not an extreme point of the whole battle group) will be checked for possibly being detected.

7) The direction will be indicated as the direction from the center of the searching group to the center of the active area (or towards the active unit itself in case of an active battle group).

D) Block diagram for Box K


In
Box K
Out
of Box K
Find extreme North, South, East West (Lat, Long) for all groups (considering extended range data)
Prog. 6
Select a pair m,n of opposing groups
Yes
Any more groups
No
Set screening flags
SCA = SCF = 0
Compare extreme points of the two groups for initial screening
Prog. 7
Yes
Any of the 2 groups active
No
SCA
1
0
SCF
1
0
Determine passive direction measurement opportunity and set values of DF(m,n) and DF(n,m)
Prog. 23
Determine pair combination
2 circles
2 rectangles
1 circle, 1 rec
BG an a circle
BG and a rectangle
2 BG's
Prog. 14
Prog. 15
Prog. 18
Prog. 20
Prog. 21
Prog. 22
Retain Df(m,n) and/or Df(n,m)

4.1.2 Step 2 - Detection Opportunity Check

a) The program arrives at this point with sets of Det(m,n) and DF(m,n) flags, equal to 0 or 1, to denote that group m has a detection opportunity against group n (if Det(m,n)=1) and possible directional measurement against group n (if DF(m,n)=1). Note that group m belongs say to side "1" and group n to side 2 (or vice versa).

b) If any of the flags (for any pair of groups) indicates the existence of a detection opportunity (i.e. Det or DF flag equal to 1) the following output routines should be automatically activated:

- A printout of a list of all pairs with a positive flag indication - both the groups having the detection opportunity and those being possibly detected. The list should include the numbers (indices) and the types of both groups, and the mode of detection, (i.e. direction to an emittance source (radar or sonar), or actual detection by radar, sonar or visual one). (Printer)
- A separate printout (on separate pages) of the above information concerning the detection relevant to each player, to be passed to the players by the umpire. However, this should be done after the umpire has already reviewed and decided upon which opportunity will lead to actual detection. (Printer)
- Eventually, when the game is going to be displayed on some computerized geographic map, an indication of the relevant pairs, by say some flashing symbols, will be useful. (Monitor display)
- The umpire may need at this point information about the groups. A full printout of positions, status, mode of operation etc. should be available. (As a matter of fact this type of printout should be possible at any point during the game progress upon the umpire's request). (Printer)

- The game step result should be fully recorded for future review, or game replay, say every 5 hours, unless a meaningful event has occurred (new instructions by the players, new information affecting the game, new groups added, positive detection) (Disc)
- In addition to the above recording, the last two steps should be temporarily stored for replay. They may be erased one by one with each additional game step except those retained for future analysis as previously mentioned. (Disc or memory)

c) In addition to the above output, a graphical display of the groups, one pair at a time, showing the geometrical intersection, will be drawn at the umpire's request for the review of the opportunity (discussed later in Sec. 3)
(Monitor display)

d) A directional measurement should be displayed as an arrow leading from the center of the detecting group towards the center (or the active particular unit) in the detected group. The direction angle β to be displayed is calculated using function 29, and then converting it to the monitor x-y display. (Monitor display)

4.2.13 Box M1 - Umpire's Considerations for Game Progress

a) At each simulation break (each time-step) (even when no detection opportunity is identified) the umpire has the opportunity for changes, additions or modifications according to his judgement and based on his perception of the game status either displayed for him on the screen or summarized in printout.

b) The printout can be obtained by the umpire upon request and should include all group locations (Lat, Long), course and speed data, and sensor status ("on" or "off"). An option for printout of partial information should be provided, such as

- Print only all blue (e.g. #1) groups or
- Print only groups No. (e.g. 3,5,7) or
- Print only SS groups or
- Print all radar operating (with radar "on") groups

c) Changes or additions by the umpire may include for example

- Change of time-step
- Addition of new battle or area groups
- Change of mission,
- Incorporation of additional (e.g. intelligence) data
- Imposition of limitations on players (e.g. [illegible]

d) If the changes require new decisions, the umpire will transfer the required data to the player (move from Box M1 back to Box S1). If the changes require only model data modification, the umpire will enter the data into the model through Box N.

e) If no changes are made, and the umpire wishes to proceed with the next game step, direct return to Box [illegible] is made.

4.2.14 Box N - Model Data Modifications/Additions

Changes, additions or modifications of data may inspire the following calculations in the model:

a) Change of simulation time-step will initiate renewed calculations in boxes H an I to reestablish group locations in the future new time-steps according to the players plans. However if the step changes are made wisely (see note below), these renewed calculations may be unnecessary. For example, if the step is shortened to 1/2 an hour, the model can still proceed with the old plans changing conditions every two steps instead of each one.
Note: In general it is recommended that the simulation time step change will be avoided. However, if needed (probably shortening it) it is desired to fix the steps in such a way that they will include previously calculated points in time (i.e. cut the time-step into 1/2, 1/3 but not 0.4 of the previous step size).

b) Addition of a battle group by the umpire's decision is unlikely. If at all, it should be done by the players decision to deploy it, i.e. through boxes E,F,G. However, addition of a group by the umpire is possible. Box N should prepare the group data for treatment as any other groups previously processed in Box H.

c) Additional data may relate to change in scenario (e.g. weather effect on radar ranges or temperature effect on sonar ranges). So far these features have not been included in the model presented here. In a general form (say by an increase or decrease of the ranges by a given factor) they are not difficult to incorporate; however the need for such refinement in the model is doubtful.

d) Note that where either the [illegible] [illegible] changed in [illegible] and/or ENCON plan calculations due to any reason (hence dictated by the players or by the umpire) the previously calculated sequences of tracks for the relevant group should be cancelled to be replaced by new ones

4.2.15 Boxes O-P - Detection Opportunity Review by the Umpire

a) Once a detection opportunity is achieved, a decision must be made by the umpire.
Note: The display will show say flashing area shapes intersecting or flashing arrows indicating directional detection.

b) The umpire's decision whether detection (rather than just an opportunity) is achieved can be based on many considerations according to his judgement:
- The probability of detection between the two concerned groups according to their types.
- The extent of the intersection, relative to the total area covered and/or patrolled by each group, and their relative courses.
- The search modes of both groups (active, passive) and the possibility of deployment of various types of sensors (towed arrays, sonobuoys etc.).
- Weather conditions specified for the game.
- The umpire's intuition and experience.
- Non-operational considerations, such as the need to dictate particular situations for practice purposes.

c) To make the decision, the umpire should have the following information:
- A full printout of all relevant pair positions, types, mode of operation, course and speed etc.
- A graphical display, obtained upon request, of the intersection yielding the detection opportunity.

d) If the umpire decides to ignore the opportunity i.e. decide that actual detection was not realized, a return to Box [illegible] is made for possible changes or for a direct return to the model to perform the next step (Box [illegible]).

8. If a detection is made directly, the [illegible] by one player or by both, information to that effect should be relayed to the appropriate player. A special printout for each player separately should be available.

4.2.6 Box Q - Detection/Kinematics Model exit

a) At this point the umpire will initiate two types of models

- Processing of the data accumulated during the game up to this point, for possible future display and analysis.
- Initialization of a second model - probably an engagement model, starting at the state obtained after a positive detection made by at least one of the players who decides to engage.

b) It is reasonable to assume that the engagement will involve only a few of the groups, surely not all the battle forces. Therefore, once an engagement is reached, the model will, by the umpire's instruction, store the present status of *all* data for restarting after the engagement model was played. Upon return from the engagement model to the kinematics/detection one, modifications should be made to account for the engagement results.

c) A return from Box Q to Box G is therefore very plausible

4.3 Subprograms, Mathematical Algorithms, Functions, Input/Output

4.3.1 Detailed explanations, Algorithms and Flow Diagrams

Note: In all calculations:

- X-Y Variables are expressed in N.M.

- Lat/Long variables (Latitude and Longitude) are expressed in angular minutes, so the say $(Lat_2 - Lat_1)$ value has a meaning of distance in N.M.

- For trigonometric functions of Lat and Long (say cos(Lat)) a conversion of the Lat or Long values into (°) or (Rad) is needed according to convenience. A reversed conversion is needed from the (°) or (Rad) values to the (min) units (to represent N.M.) after say a $\cos^{-1}(\ldots)$ operation.

- Long=0 at Greenwich, assuming positive values going eastbound and negative values going westbound.

- Lat=0 at the equator. Positive values are used in the north hemisphere and negative at the south one.

Program 1 Coordinate Conversion from the Game (X-Y) into (Lat,Long) Values

Input: (x,y) - point to be converted (in N.M.)
Equivalent (x_0,y_0) and $(Lat_0,Long_0)$ as a reference point.

Output: (Lat,Long) (in angular minutes)
Note: E. Long is positive, W. Long is negative.
N. Lat is positive, S. Lat is negative.

Algorithm: $Lat = Lat_0 + (y - y_0)$
$Long = Long_0 + (x - x_0)/\cos(Lat/60)$

Program 2 Coordinate Conversion from the (Lat,Long) into the Game (X-Y) Values

Input: (Lat,Long) of point to be converted (in angular minutes).
Equivalent (x_0,y_0) and $(Lat_0,Long_0)$ as a reference point.
Note: see note in Program 1.

Output: (x,y) (in N.M.)

Algorithm: $y = y_0 + (Lat - Lat_0)$
$x = x_0 + (Long - Long_0)\cos(Lat/60)$

Program 3 Determination of the 4 Rectangle Corners Using the Sides a and b

Input
x_0, y_0 - the rectangle center
a,b - the rectangle sides (in m)
Co - course (°)
α - the angle of "a" relative to the course (positive clockwise)

Output (x_i, y_i) i = 1 to 4

Algorithm (see drawing)

$\theta = Co + \alpha$

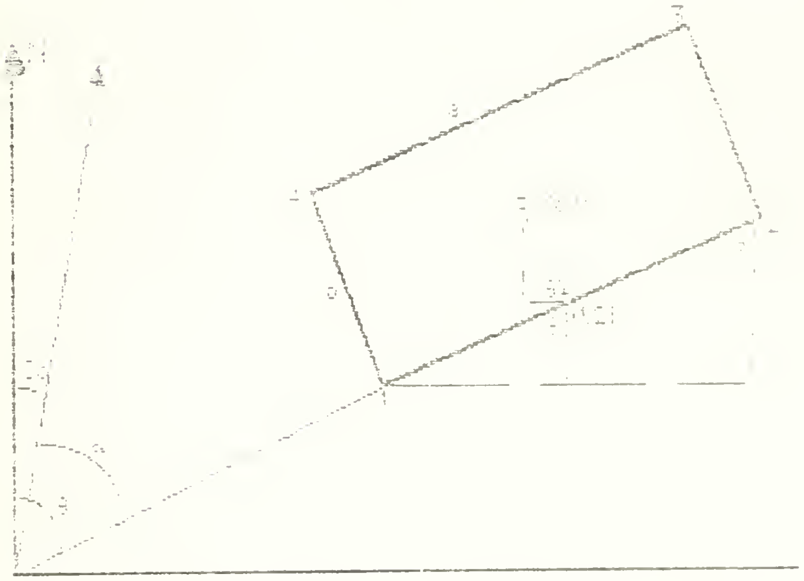
$x_p = x_0 + (b/2)\cos(\theta)$; $y_p = y_0 - (b/2)\sin\theta$

$x_1 = x_p - (a/2)\sin\theta$; $y_1 = y_p - (a/2)\cos\theta$

$x_2 = 2x_p - x_1$; $y_2 = 2y_p - y_1$

$x_3 = 2x_0 - x_1$; $y_3 = 2y_0 - y_1$

$x_4 = 2x_0 - x_2$; $y_4 = 2y_0 - y_2$



Program 4 Determination of the Remaining 2 Corners of a Rectangle Given 2 Corners and a Point on the Opposite Side

Input: (x_1,y_1) (x_2,y_2) - 2 rectangle corners.
(x_{34},y_{34}) - a point on opposite side

Output: $(x_i,y_i)_{i=3,4}$
a, b - rectangle side length (NM)

Algorithm: (Refer to drawing).

1) Calculate "a" - the distance from point 1 to point 2 (use Function 24).
2) Determine line $\overline{1,2}$ coefficients A,B,C (use Function 21)
3) Calculate the distance d (=b) of point (x_{34},y_{34}) from the line $\overline{1,2}$ (use Function 22).
(note that in function 22 $(A^2+B^2=a^2)$ is already calculated in step 1).
4) Calculate k - the distance of point (3,4) to 2 (use Function 24).
5) Perform the following calculations:

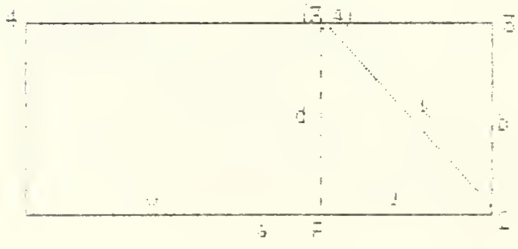
$u = \sqrt{k^2-b^2}$, $v = a - u$

$x_p = x_1 - (x_2 - x_1)V/a$, $y_p = y_1 - (y_2 - y_1)V/a$

$E_x = x_{34} - x_p$, $E_y = y_{34} - y_p$

$x_3 = x_2 + E_x$, $y_3 = y_2 + E_y$

$x_4 = x_1 + E_x$, $y_4 = y_1 + E_y$



Program 5 Calculation of Corners of an Extended [illegible] Active Sensors

Input

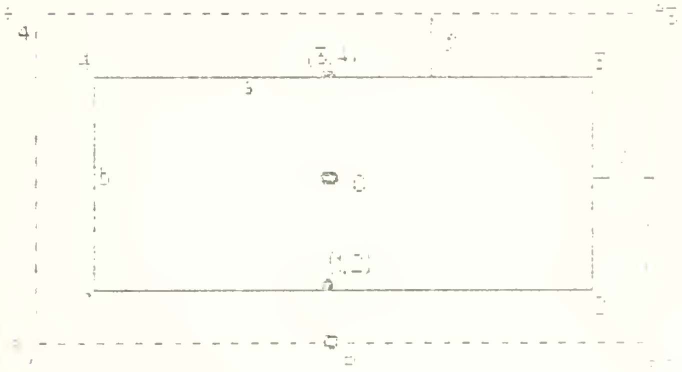
(x_c, y_c) - rectangle center
$(x_i, y_i)_{i=1 \text{ to } 4}$ the 4 rectangle corners
a, b - the rectangle sides $(a = \overline{12}, b = \overline{23})$
δ - the extension (equal to the [illegible] range).

Output

$(x_{ei}, y_{ei})_{i=1 \text{ to } 4}$ - extended corners

Algorithm

(Refer to drawing)
$x_o = (x_1 + x_2)/2$
$y_o = (y_1 + y_2)/2$
$x_p = x_c - (x_o - x_c)(b + 2\delta)/b$
$y_p = y_c - (y_o - y_c)(b + 2\delta)/b$
$\{x_{ei} = x_p + (x_i - x_o)(a + 2\delta)/a\}_{i=1 \text{ to } 2}$
$\{y_{ei} = y_p + (y_i - y_o)(a + 2\delta)/a\}_{i=1 \text{ to } 2}$
$\{x_{ei} = 2x_c - x_{e(i-2)}\}_{i=3 \text{ to } 4}$
$\{y_{ei} = 2y_c - y_{e(i-2)}\}_{i=3 \text{ to } 4}$



Program 6 Determination of the Extreme Latitude and Longitude Points in a Group

Input: For a rectangular area group and a battle group
(lat$_i$,Long$_i$)$_{i=1 \text{ to } n}$ (n=4 for a rectangle, n=[illegible] for a Battle Group). The values are those of the basic area - unextended)

For a circular area group:
Lat$_0$, Long$_0$, R$_0$ (unextended)

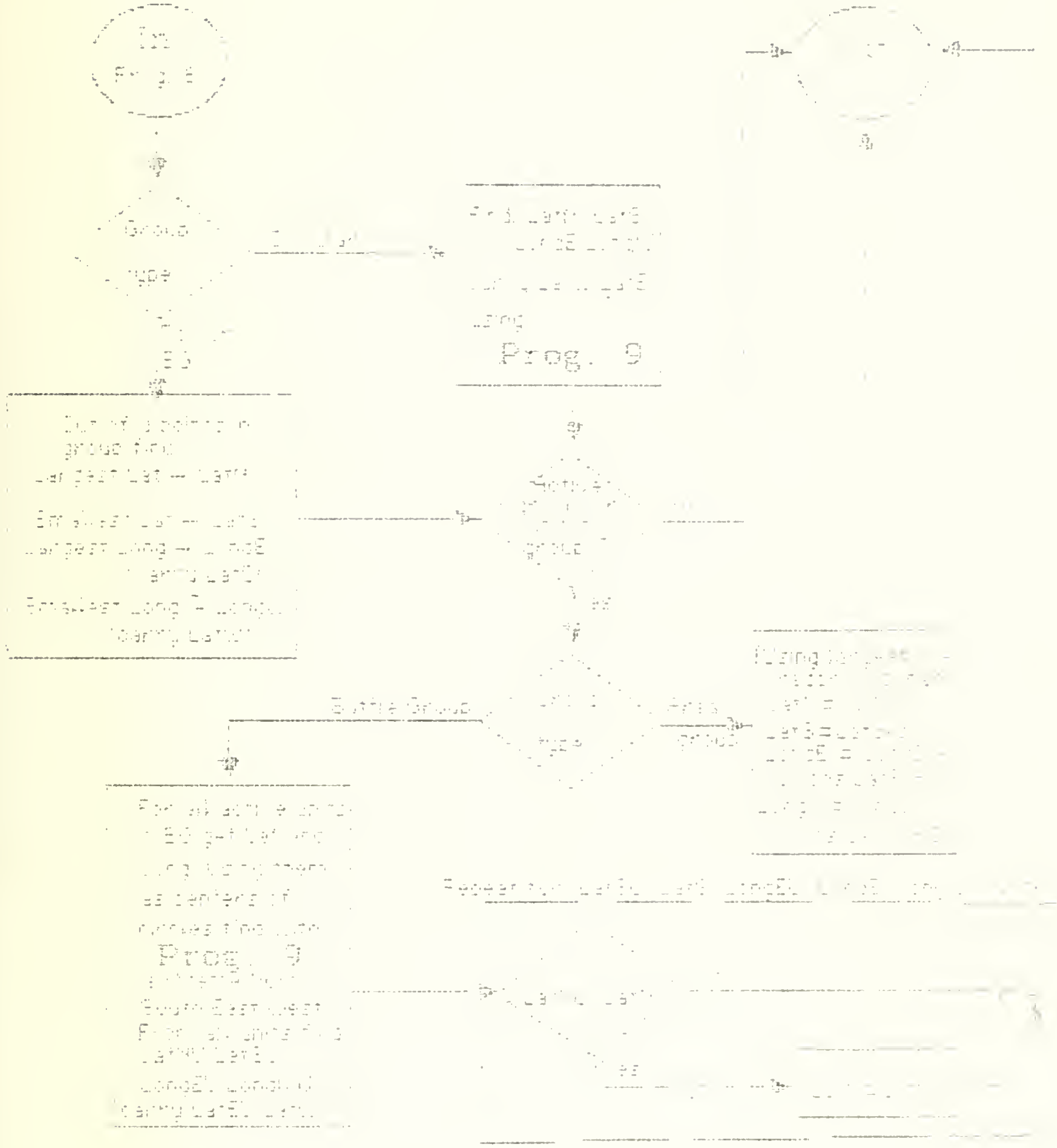
For an active area group (rectangular or circular): Largest δ (range extension) among the active extensions (say radar or sonar

For an active battle group: maximal δ for each of the active units in the group

Output: Extreme North, South, East and West points of the group, including extensions, expressed as LatN, LatS, LongE, LongW (carrying LatE and LatW).

Note: no need to carry LongN and LongS values.

Algorithm: See flow diagram.



Group
type
Prog. 9
group
Battle Group
group
Prog. 9

Program 7 Comparison of Extreme Points of Two Groups for Screening Purposes

(Both active and passive groups)

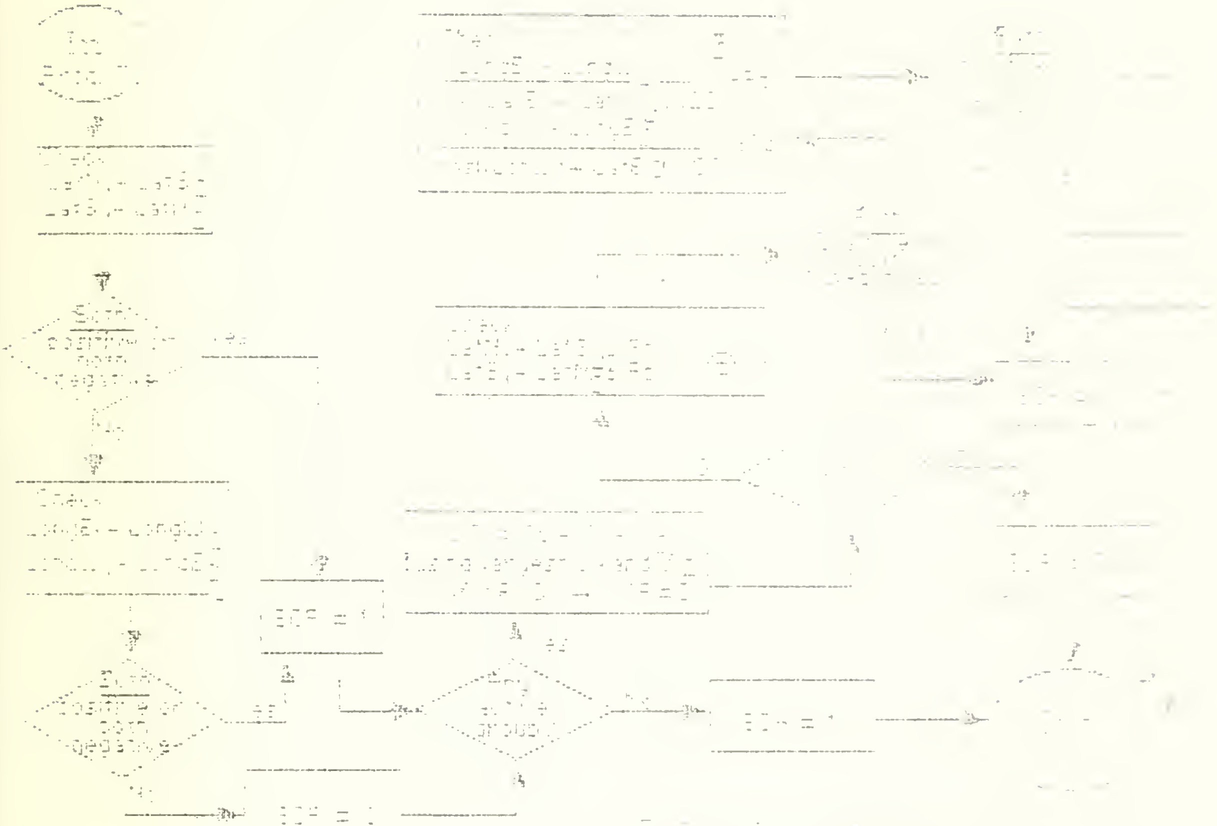
Input: LatN, LatS, LatE, LatW, LongE, LongW for both groups.

δ_1, δ_2 - the range extensions of the two groups

R_p - range of a passive detection of an active target, e.g. by ESM.

Output: SCA = 0 or 1; SCP = 0 or 1 - screening flags (SCA for active target, SCP for passive one). The values 1 and 0 indicate that the pair considered is screened or not screened respectively from further analysis of detection opportunity

Algorithm: See flow diagram.



[illegible]

Program 8 Calculation of of the Parameters A,B,C of a Line Passing through Two Points

Input: $(x_i,y_i)_{i=1\ to\ 2}$ – the two points through which the line passes.

Output: The constants A, B, C, of the straight line $Ax + By = C$

Algorithm:

$A = y_2 - y_1$

$B = x_1 - x_2$

$C = x_1y_2 - x_2y_1 - 2x_1y_1$

Program 9 Finding Extreme Points on a Circle

Input:
$Lat_0, Long_0$ – circle center coordinates
R_0 – circle radius (in N.M.).

Output: LatN, LatS, LongE, LongW (carrying LatE and LatW – extreme points (no need to carry LongN and LongS values).

Algorithm:

$LatN = Lat_0 + R_0$

$LatS = Lat_0 - R_0$

$LatE = Lat_0$

$LongE = Long_0 + R_0/\cos(Lat_0/60)$

$LatW = Lat_0$

$LongW = Long_0 - R_0/\cos(Lat_0/60)$

Program 10 Rotation of a Point P around a Point [illegible]

Input: $Lat_c, Long_c$ - center of rotation

$Lat_p, Long_p$ - coordinates of point P to be rotated

α (+=CCW) - angle of rotation

Output: $Lat_r, Long_r$ - (r being the point P after rotation)

Algorithm: See auxiliary drawing below.

$\Delta y = Lat_p - Lat_c$

$\Delta x = (Long_p - Long_c)\cos((Lat_c + Lat_p)/2)$

$R = \sqrt{\Delta x^2 + \Delta y^2}$

$\theta_p = Tan^{-1}(\Delta y/\Delta x)$

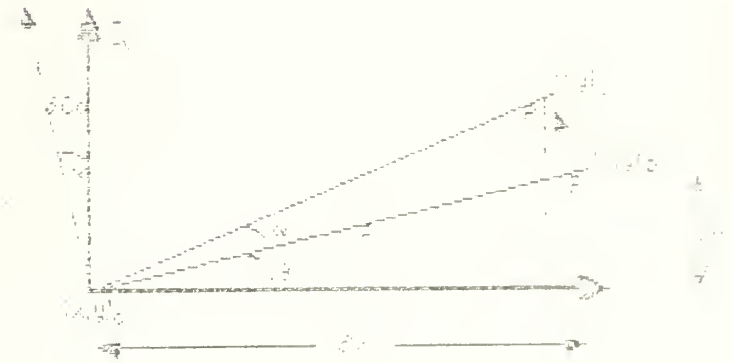
$\theta_r = \theta_p - \alpha$

$\Delta x_r = R_c \cos\theta_r$

$\Delta y_r = R_c \sin\theta_r$

$Lat_r = Lat_c + \Delta y_r$

$Long_r = Long_c + \Delta x_r/\cos((Lat_c + Lat_r)/2)$



Program 11 Track Plans 2 and 3, Based on Destinations, and Speeds or Time of Arrivals Respectively

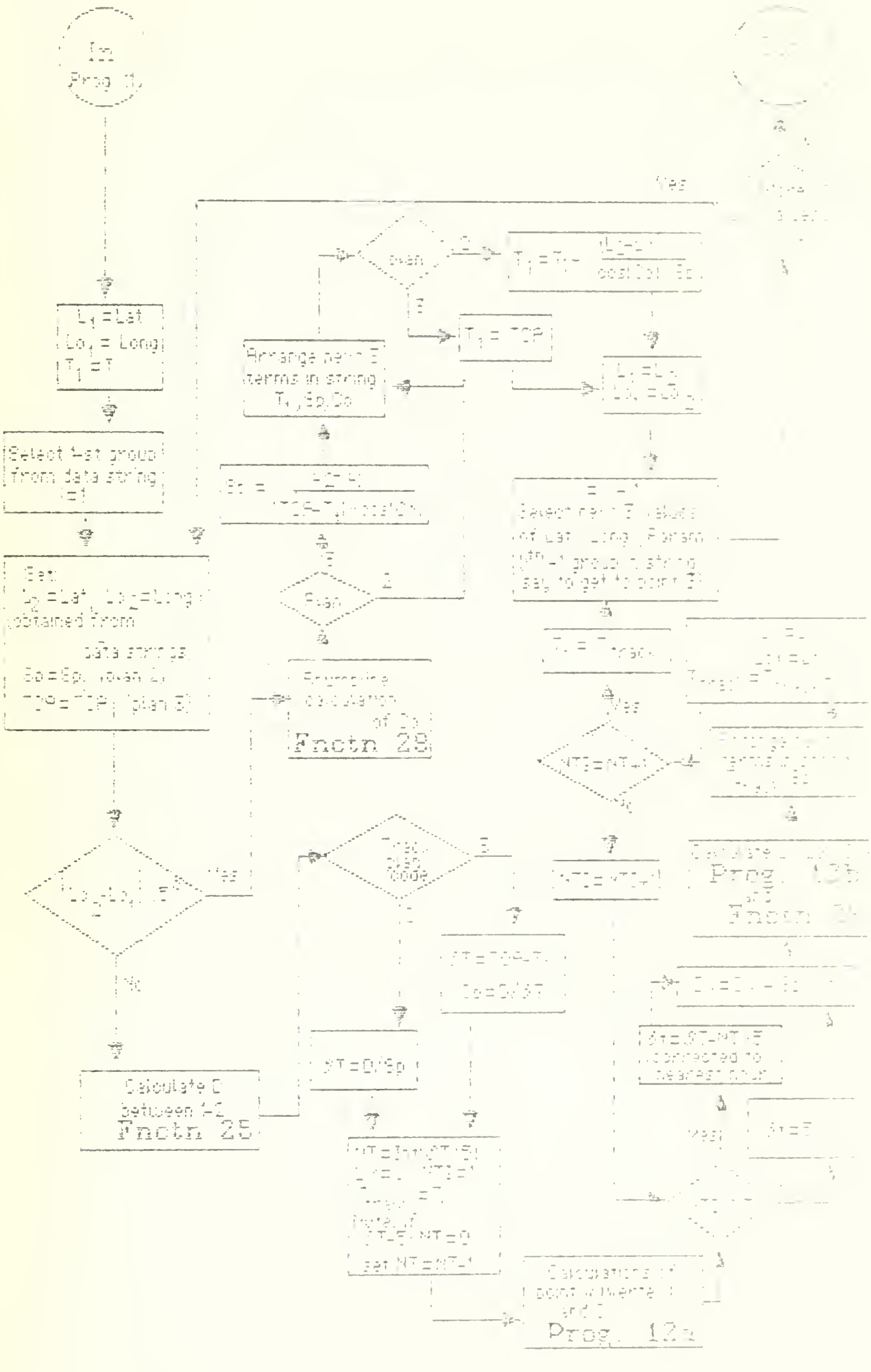
Input. A sequence of $Lat_1, Long_1, param_1, Lat_2, Long_2,$ $param_2, \ldots\ldots$
where $param_i = Sp_{i-1}$ for plan code 2,
and $param_i = TOA_i$ for plan code 3,
and all times are relative to T_0

Output A sequence of: $T_0, Sp_0, Co_0, T_1, Sp_1, Co_1, \ldots\ldots$
(approximating a great circle).

Note The T_i will generally be in 5 hours increments except for cases where course and/or speeds must be changed at different times by players instructions.

Procedure: Use great circle calculations to find points on the track for course changes (usually 5 hours apart). Once the points are determined, use local rhumbline calculations for constant course track from one point to another (see auxiliary drawing in Prog 12).
Note: In some cases where the change in longitude between the initial point and the destination is smaller than 5°, the great circle calculations are not necessary.

Algorithm: See flow diagram.


In
Prog 11
L1 = Lat
Lo1 = Long
T1 = T
Select 1st group from data string
Get
L2 = Lat, Lo2 = Long
obtained from
data strings
Sp (plan 1)
TCP (plan 2)
Yes
No
Calculate D between 1-2
Fnctn 25
Plan
Sp =
Arrange next
terms in string
T1, Sp, Co
Yes
Plan
T1 = TCP
Select next
of Lat, Long
group in string
NT = NT + 1
Summing
calculation
of Co
Fnctn 28
Plan
Code
ST = D/Sp
Co = D/ST
set NT = NT + 1
Prog 12b
Fnctn
corrected to
nearest hour
Yes
Calculations
Prog. 12a

Program 12 Calculation of Lat and Long of a Point X on a Great Circle Track, to Find the Course Towards It

(Divided into programs 12a and 12b)

Input: D — great circle length (from point 1 to 2)
L_1, Lo_1 — Lat and Long at point 1 (in °)
L_2, Lo_2 — Lat and Long at point 2 (in °)
D_x — segment length (in NM) from point 1 to point x (input to Prog 12b)

Output: C_x — the course from point 1 towards point x
Note: Point 1 and x change continuously, i.e. point 1 is at the start of the great circle track. Later it assumes step by step the coordinate values of the previously calculated point x along the great circle.

Algorithm: (in two parts)

Program 12a

(1) Convert D_{NM} equivalent in (min) to (°) by $D/60$ to get $D(°)$ and calculate $\sin D(°)$
(2) $\sin(C) = \sin(Lo_2 - Lo_1)\cos L_2 / \sin(D)$
$\cos(C) = \sqrt{1 - \sin^2(C)}$
(3) $\cos(L_v) = \cos(L_1)\sin(C)$
$\sin(L_v) = \sqrt{1 - \cos^2(L_v)}$
(4) $\sin(DLo_v) = \cos(C)/\sin(L_v)$
Find $DLo_v(°)$ as $\sin^{-1}(\;)$ (needed for step 10)
(5) D_v (in NM) $= 60\sin^{-1}(\cos L_1 \sin(DLo_v))$

Program 12b

(6) D_{vx} (in NM) $= D_v - D_x$ (can be negative)
(7) Convert D_{vx} from (NM) to (°) by $D_{vx}/60$ to get $D_{vx}(°)$ and calculate $\cos(D_{vx})$
(8) $\sin L_x = \sin L_v \cos(D_{vx}(°))$
Find $L_x(°)$ as $\sin^{-1}(\;)$ (needed for step 11)

[illegible]

9) [illegible]

Find [illegible]

10) [illegible]

11) [illegible]

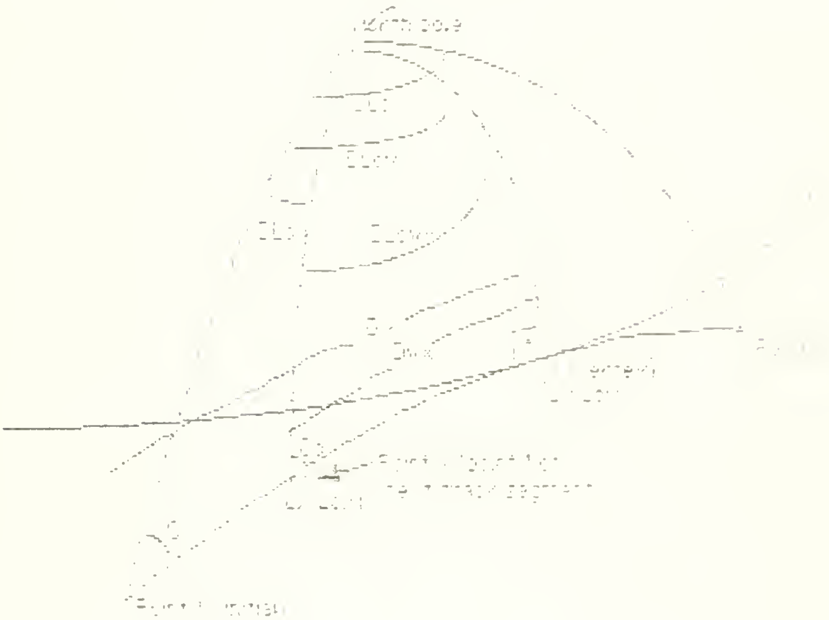
[illegible]

12) use Function 26 to calculate [illegible] thumbline step from point [illegible]

Note point [illegible] changes along [illegible]

assume coordinate values [illegible]

calculated point V.



Program 13 One Step Rhumbline Advancement

Input: Lat_1, $Long_1$ - initial position coordinates
Sp, Co - speed and course for advancement,
δT - simulation time-step.

Output: Lat_2, $Long_2$ - position coordinates at the end of the step.

Algorithm:

```
Lat_2 = Lat_1 + Sp×δT×cos(Co)
Temp = cos((Lat_1+Lat_2)/(2×60))
Long_2 = Long_1 + Sp×δT×sin(Co)/Temp
```

Program 14 Checking for Possible Detection Opportunity between Two Circular Area Groups (e.g. Air or Sea)

Input: Lat_m, $Long_m$, Lat_n, $Long_n$ - coordinates of centers of the two groups

R_m, R_n - Radii of the two circles

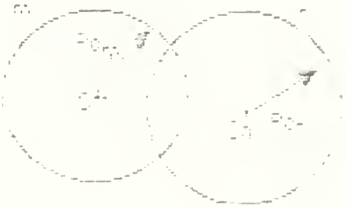
DR_m, DS_m, DR_n, DS_n - range extent of the best active sensors (e.g. radar and sonar) need to be found only if radar and/or sonar for either side are in active position.

Output: An indication of Detect/no Detect will be given: 0 or 1 to designate a detection opportunity. A detection flag indicates that group m has a detection opportunity against group n.

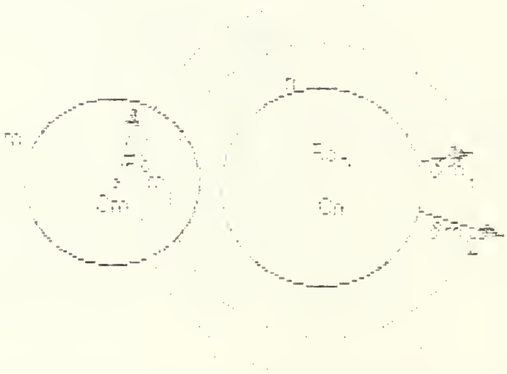
Procedure: The procedure will be based on three possible combinations of the two groups:

Combination 1 - both groups passive

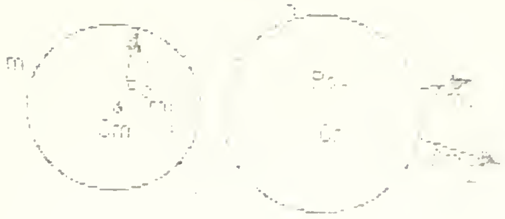
The intersection points of the two circles (if they intersect) [illegible] $R_m + R_n$.



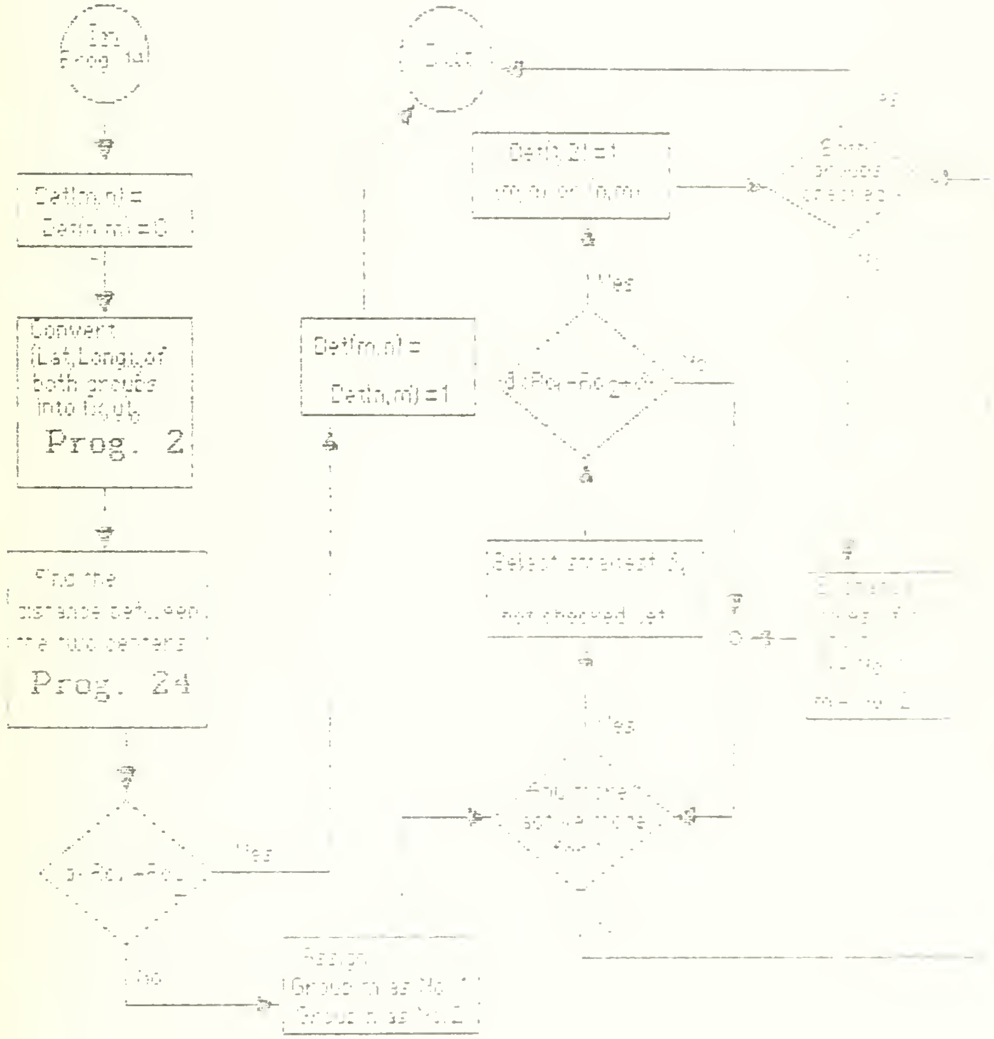
Combination (2) - one group (say 'm') passive, and the other (say n) is active.
Note: there can be more than one extended range, say for radar and sonar if both are active. In this case, in addition to the intersection of the two basic circles (as in combination 1 above), one must check also the intersection between the extended range(s) circle of n (the searcher) and the basic circle of m (the target). This is done for each extended range in the order of increasing range.



Combination (3) - both groups active.
The procedure is now carried out as in combination 2 above, for each group separately, each as an active searcher against the basic area of his opponent as a target. Note that the basic circles intersection does not have to be checked twice.
In the example illustrated in the drawing in the next page a case is shown where n has a detection opportunity against m but not vice versa.



Algorithm: See flow diagram



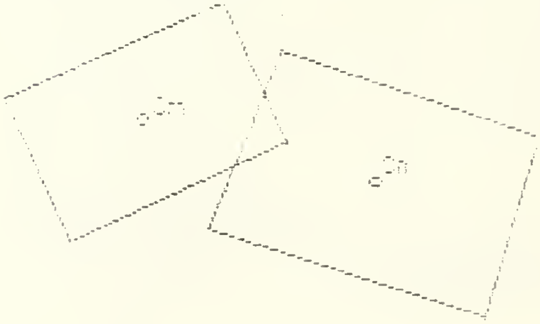
Program 15 Checking for Possible Detection Opportunity between Two Rectangular Area Groups (say No. m and n)

Input: $\{Lat_{mi}, Long_{mi}\}_{i=1 \text{ to } 4}$ – rectangle m coordinates.
$\{Lat_{ni}, Long_{ni}\}_{i=1 \text{ to } 4}$ – rectangle n coordinates
With active radar (R) or sonar (S) need also
$\{LatR_{mi}, LongR_{mi}, LatS_{mi}, LongS_{mi}\}_{i=1 \text{ to } 4}$
$\{LatR_{ni}, LongR_{ni}, LatS_{ni}, LongS_{ni}\}_{i=1 \text{ to } 4}$
Note: the procedure does not require the two rectangle center coordinates.

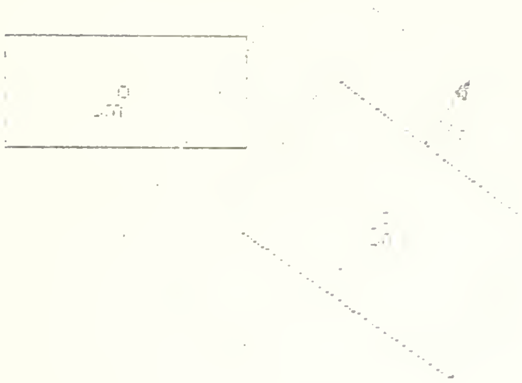
Output: An indication of Det(m,n) or Det(n,m) equal to 0 or 1 to designate a detection opportunity. A Det(m,n)=1 flag indicates that group m has a detection opportunity against group n.

Procedure: The procedure has in principle the same three combinations as for the case of two circular areas (see program 14).

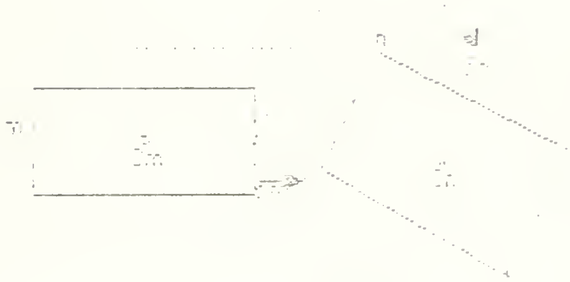
Combination (1) – two passive groups



Combination (2) - one [illegible]
group



Combination (3) - two active groups

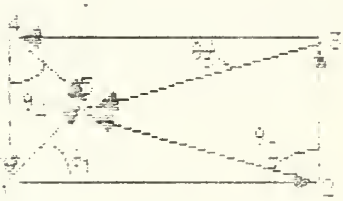


The four types of intersection checks [illegible]
be:

(i) Check that the intersection between the
their rectangles

5. For each active group, check the intersection between its extended area(s) and the opponent basic area (again, two extensions for radar and sonar may exist).

Checking the intersection between two rectangles will be based on the property that each of the angles θ_i between the 4 pairs of vectors (as shown in the diagram) is smaller than 90° (or alternatively – $\cos(\theta_i) > 0$).



Note in the drawing that for point 5 to be inside the rectangle (1,2,3,4) all the four values of $\cos(\theta_i)_{i=1 \text{ to } 4}$ must be positive. Any point (say 5') outside the rectangle will yield at least one θ value larger than 90° with a negative $\cos\theta_i$.
The basic formula for the angle between two vectors $\overline{a}$ and $\overline{b}$ is

$$\cos(\theta_{ab}) = (\overline{a} \cdot \overline{b}) / (|a| \cdot |b|)$$

where $\overline{a}$ and $\overline{b}$ are any two vectors in the plane.
The 4 pairs of vectors to be checked are

(1) $(\underbrace{x_5-x_1}_{x_a}, \underbrace{y_5-y_1}_{y_a})$ with $(\underbrace{x_4-x_1}_{x_b}, \underbrace{y_4-y_1}_{y_b})$ to yield θ_1

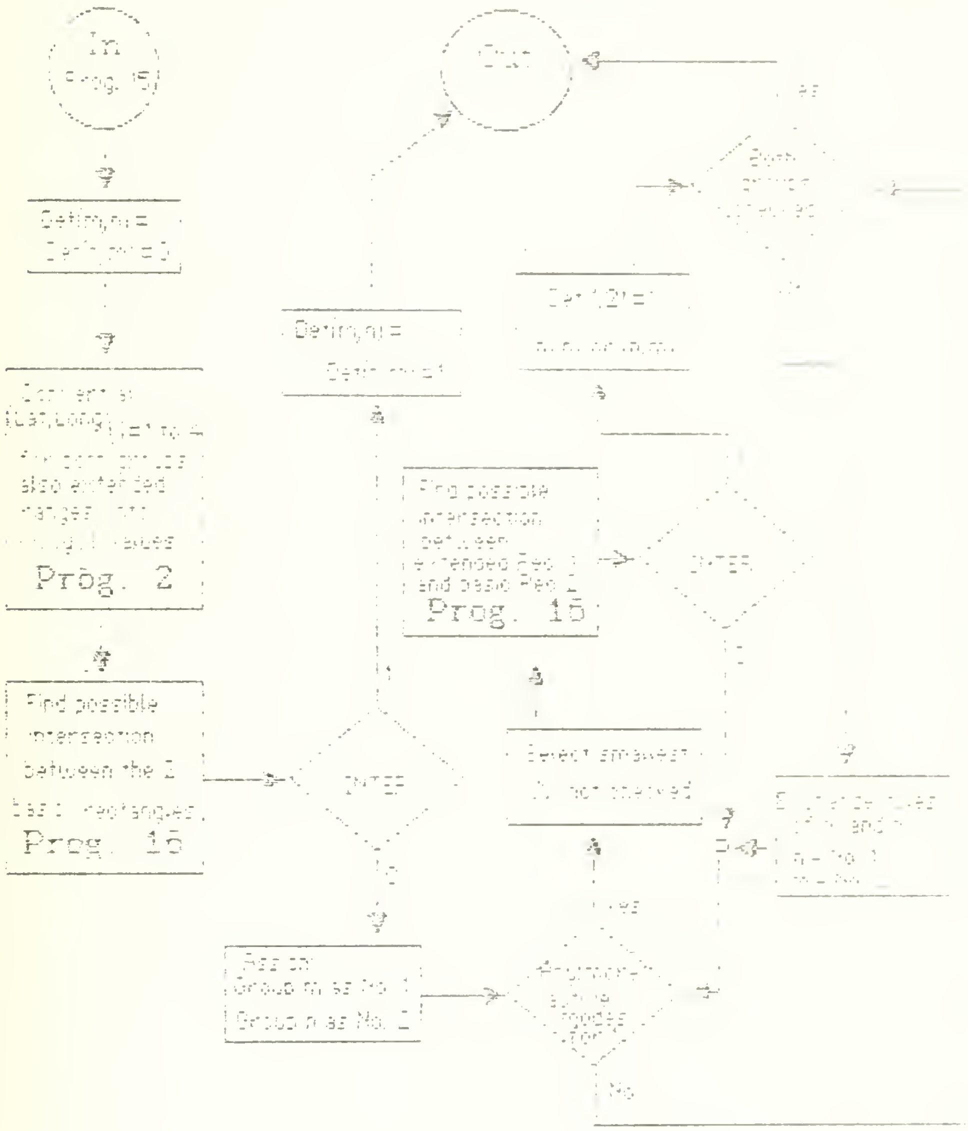
(2) (x_5-x_2, y_5-y_2) with (x_1-x_2, y_1-y_2) to yield θ_2

(3) (x_5-x_3, y_5-y_3) with (x_2-x_3, y_2-y_3) to yield θ_3

(4) (x_5-x_4, y_5-y_4) with (x_3-x_4, y_3-y_4) to yield θ_4

The procedure will therefore check one by one each corner of a rectangle to find whether it is inside the other rectangle, proceeding thereafter to check the second rectangle corners. Once a corner is found to be inside the other, coincidence is established and there is no need to check the remaining corners.

Algorithm: See flow diagram

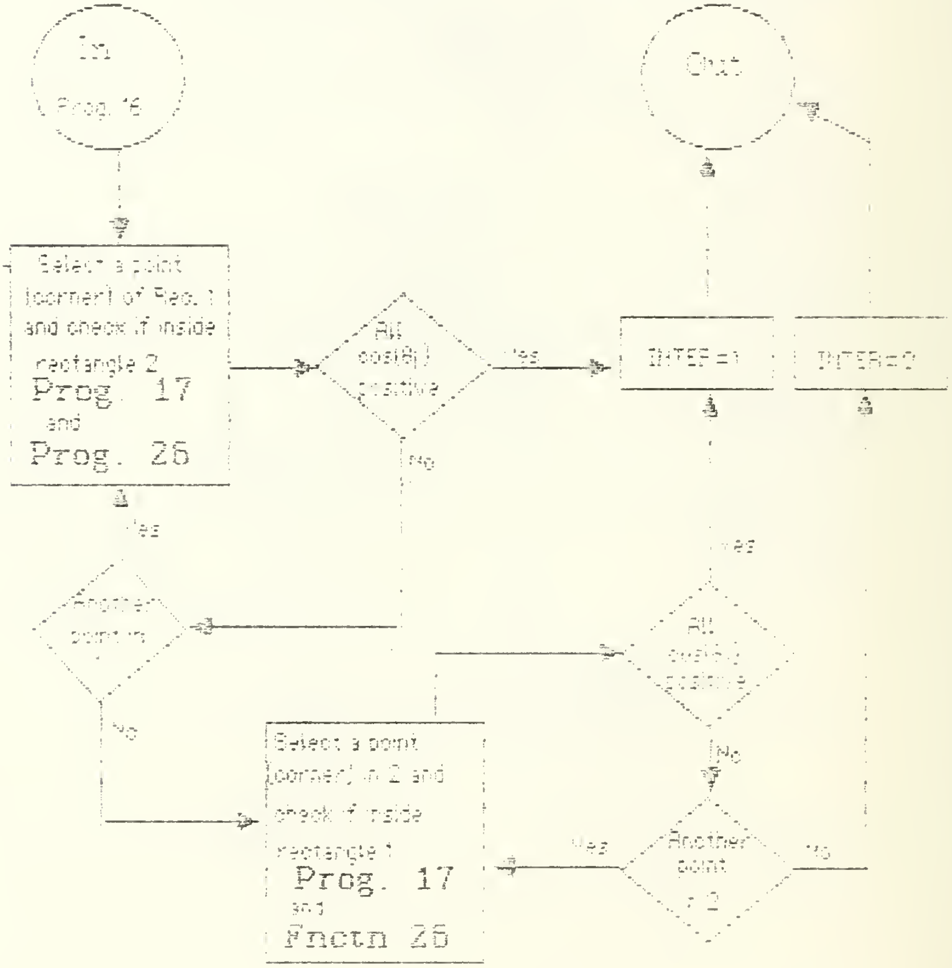

In
Prog. 15
Out
Define n =
Define m = 1
Convert to (Lat,Long) i = 1 to 4
The corners are also extended ranges
Prog. 2
Find possible intersection between the 2 basic rectangles
Prog. 15
INTER
Define n =
Define m = 1
Find possible intersection between extended Rec 1 and basic Rec 2
Prog. 15
INTER
Select smallest not checked
Yes
No

Program 16 Finding an Intersection between Two Rectangles

Input: $\{x_i, y_i\}_{i=1 \text{ to } 4}$ corners of the first rectangle
$\{X_j, Y_j\}_{j=1 \text{ to } 4}$ corners of the second rectangle

Output: An indication for an intersection by INTER=1,0 if an intersection was found or not found respectively.

Algorithm: See flow diagram.



Program 17 Finding whether a Point P is Inside a Given Rectangle

Input x_0, y_0 – coordinates of a given point

$x_1, y_1, \ldots, x_4, y_4$ – rectangle corners coordinates

Output $\cos\theta_{21}$, $\cos\theta_{32}$, $\cos\theta_{43}$, $\cos\theta_{14}$

Algorithm $x_a = x_0 - x_1$

$y_a = y_0 - y_1$

$x_b = x_2 - x_1$

$y_b = y_2 - y_1$

Use Function 26 to calculate $\cos\theta_{21}$

Repeat calculations for the pair 2-3, [illegible] respectively, to calculate $\cos\theta_{32}$, then $\cos(\theta_{43})$ and $\cos\theta_{14}$ in the same manner.

Note. Once any $\cos\theta$ is found to be negative there is no need to proceed with the rest $\cos(\theta)$ calculations.

Program 18 Checking for Possible Detection Opportunity between a Rectangular (say m) and a Circular (say n) Area Groups

Input: $\{Lat_{mi}, Long_{mi}\}_{i=1 \text{ to } 4}$ - rectangle corner coordinates

$Lat_{cn}, Long_{cn}, R_{cn}$ - circle center coordinates and radius.

If rectangular area group is active (radar and/or sonar), input also:

$\{LatR_{mi}, LongR_{mi}\}_{i=1 \text{ to } 4}$ and/or $\{LatS_{mi}, LongS_{mi}\}_{i=1 \text{ to } 4}$

If circular area group is active, need also ΔR_n and/or ΔS_n.

Output: An indication of Det(m,n) and Det(n,m) equal to 0 or 1 to designate a detection opportunity. (Det(m,n)=1 flag indicates that group m has a detection opportunity against group n.)

Procedure: Just as in programs 14 and 15, there are three combinations:

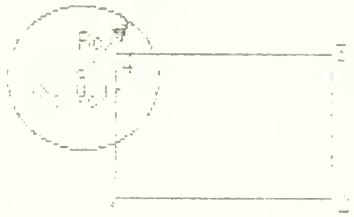
Combination (1) - an intersection between the two basic areas (two passive groups).

Combination (2) - an intersection between an extended area (an active group) and the opponent's basic area (passive group). Each extended area (radar and sonar) of the active group must be checked.

Combination (3) - for two active groups, the calculations as in combination 2 must be repeated for each group separately except for the intersection of the basic areas that need not be repeated twice.

However, for a programmer, it is [illegible] be found whether the intersection between a rectangle with 4 known corners, and a circle with known center and radius. The procedure will consist of the following stages:

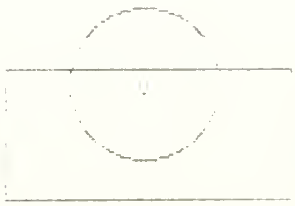
Stage 1: Find whether any of the rectangle corners is inside the circle.



Method: Check sequentially for all the four corners distance d from the circle center (using Function 24).

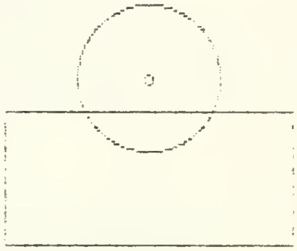
Note: Once a distance $d \le R_0$ is found, i.e. the corner is inside the circle, a detection opportunity is identified, and there is no need to further check for the rest of the corners, nor is it necessary to proceed with stages 2 and 3. If no detection opportunity is found after checking all 4 corners, proceed to stage 2.

Stage 2: Check if the circle center is inside the rectangle.



Method: Use Program 17 and Function 26 finding a point to be inside a rectangle.

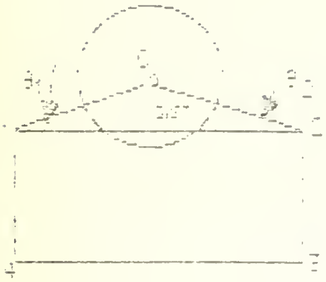
Note. If the center is found inside the rectangle, a detection opportunity is identified; no need to proceed to stage 3. However, if the center is not inside the rectangle there is still one case in which detection opportunity can still exist as shown in the next diagram. Proceed to stage 3.

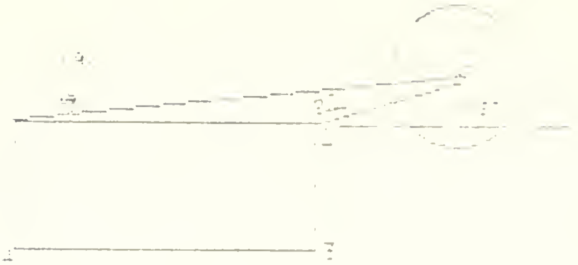


Stage 3: Check whether the circle is close enough to one of the rectangle sides.

Method: For each pair of adjacent rectangle corners, defining a straight line, find the line's distance to the circle center using Program 8 and Function 27.

Note: If the distance d is found to be $d > R_d$ for all 4 lines, discard the two groups from any additional analysis. There is no detection opportunity. However, if a distance $d < R_d$ is found, check which of the two relative positions of the two groups (as shown in the following drawing in the next page) exists.





In the left drawing there is a [illegible] opportunity exists: both angles θ_1 = ∠ 012 and θ_2 = ∠ 021 are acute. In the right drawing one of the angles, for example θ_1 = ∠ 021 [illegible] obtuse angle [illegible] the circle intersects the [illegible] outside the rectangle. [illegible] the appropriate angles [illegible] function 2b. The two [illegible] are 01 and 02 for the θ_1 [illegible] 02 and 01 for the θ_2 [illegible] follows:

[illegible]

[illegible]

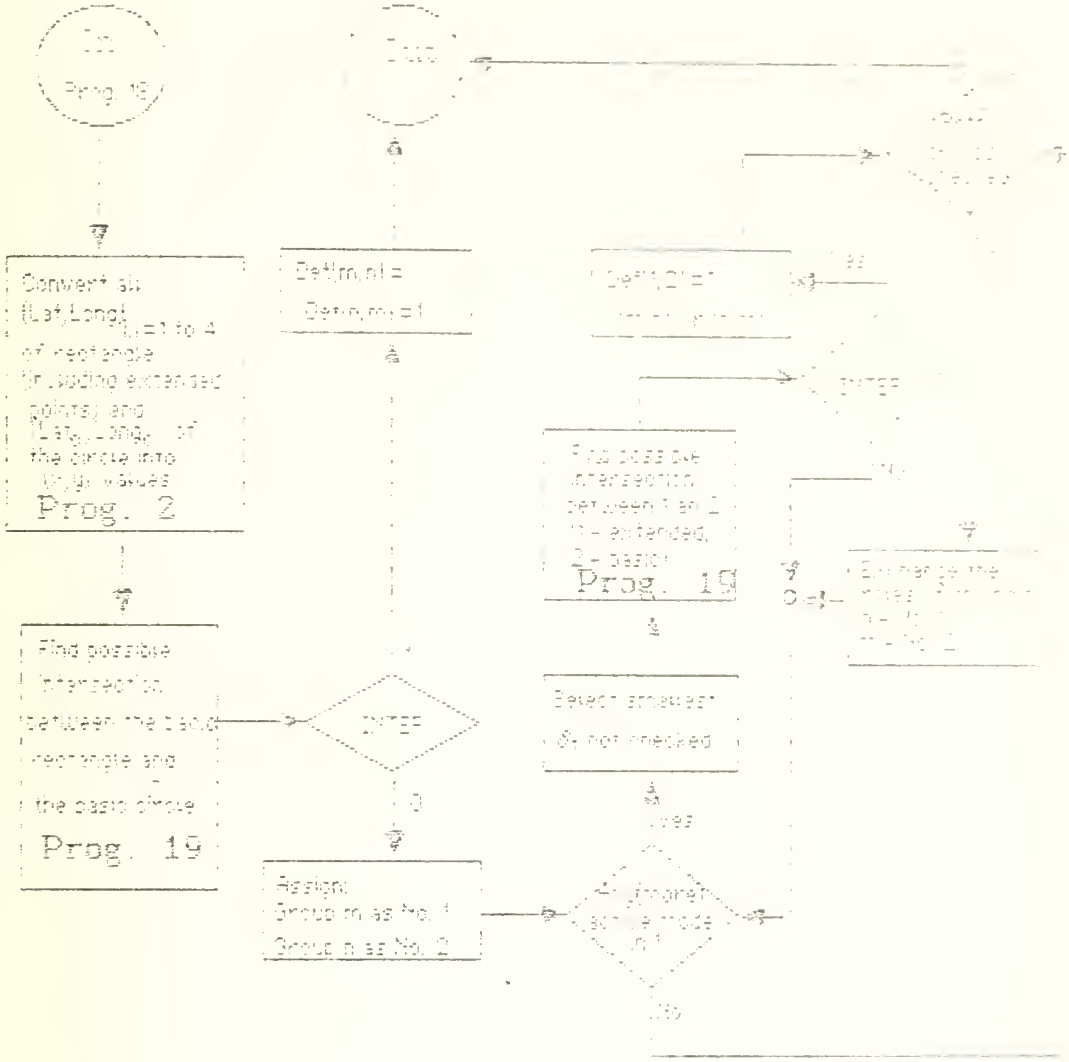
Once both angles found to be acute, there is no need for any further [illegible] detection opportunity [illegible] angle is obtuse [illegible] corner line check [illegible] stage 3.

Note There are only very few cases in which stage 3 will be reached for the following reasons:

- The screening process by checking the latitudes and longitudes [illegible] (see Program 7) [illegible] for intersection analysis in most game steps. Only when the two groups are close to each other and detection opportunity is feasible the intersection analysis is made.

- Furthermore, the nature of the screening process will leave only few cases where there still is no intersection (i.e. no detection opportunity).
- Only when there is no intersection between the circle and the rectangle, must the model go fully through all three stages. If intersection exists, the chances are it will be identified by the first two stages and only in a very few cases the third (and the more complex) stage will be needed for the identification.

Algorithm: See flow diagram.



Note the similarity to program 15 except for the use of a circle and a rectangle instead of two rectangles using therefore program 19 instead of 15 for the intersection check.

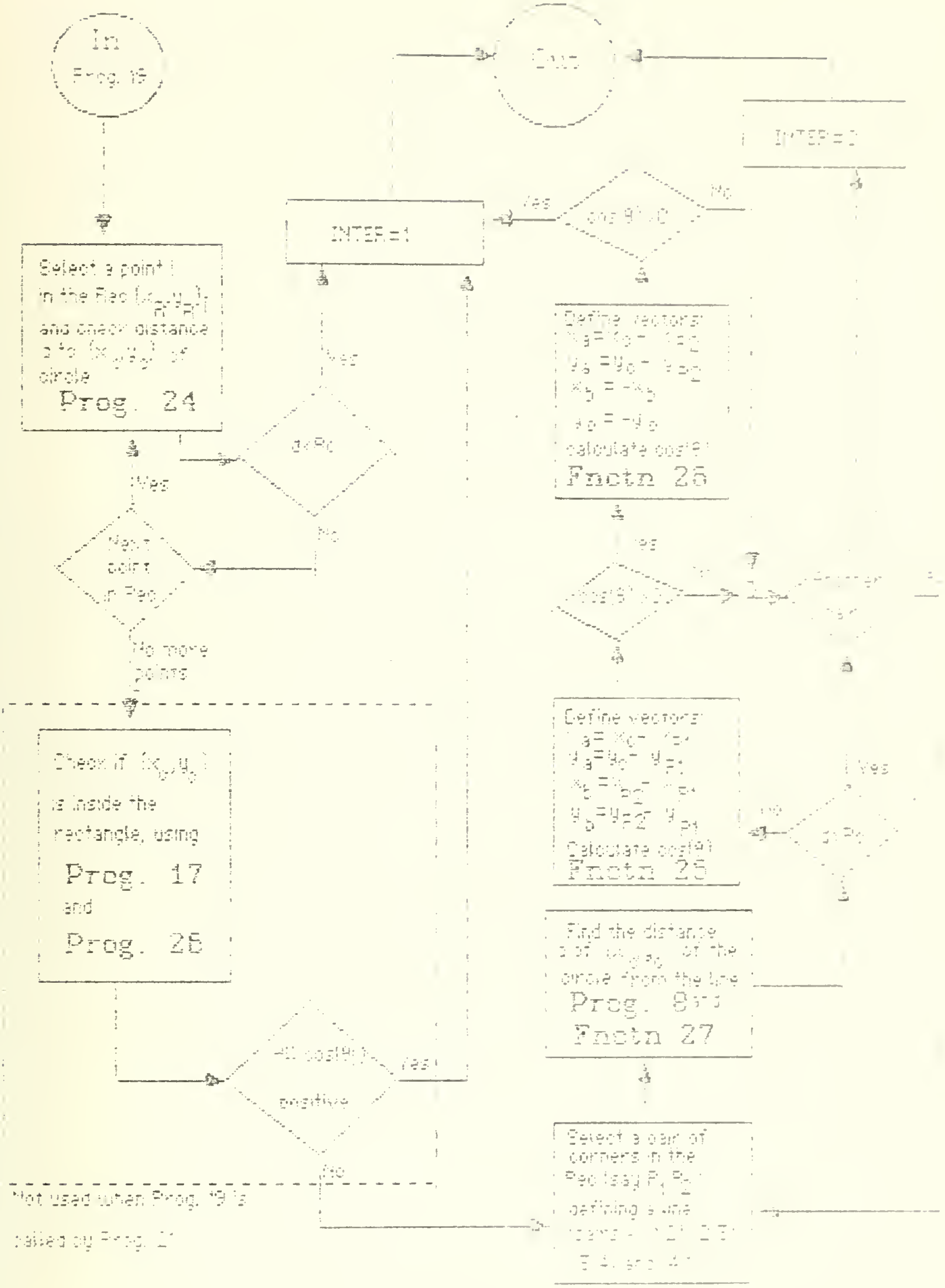
Program 19 Finding the Intersection between a Rectangle and a Circle

Input: (x_i, y_i), $i = 1$ to 4 - rectangle corner coordinates
(x_c, y_c) - circle center coordinates
R_c - circle radius.

Output: An indication of an intersection by an INTEGER flag to indicate whether an intersection was found or not found respectively.

Note: An index "R" will be used for the rectangular group and an index "C" for the circular one.

Algorithm: See flow diagram.


In
Prog. 19
Out
Select a point i in the Rec (x_{Pi}, y_{Pi}) and check distance d to (x_c, y_c) of circle
Prog. 24
Yes
Next point in Rec
No more points
d<Rc
Yes
No
INTER=1
Yes
$\cos\theta < 0$
No
INTER=0
Define vectors:
$x_a = x_c - x_{P2}$
$y_a = y_c - y_{P2}$
$x_b = -x_b$
$y_b = -y_b$
calculate $\cos\theta$
Fnctn 25
Yes
$\cos\theta < 0$
No
Define vectors:
$x_a = x_c - x_{P1}$
$y_a = y_c - y_{P1}$
$x_b = x_{P2} - x_{P1}$
$y_b = y_{P2} - y_{P1}$
Calculate $\cos\theta$
Fnctn 25
Yes
No
d>Rc
Check if (x_c, y_c) is inside the rectangle, using
Prog. 17
and
Prog. 26
All $\cos\theta$ positive
Yes
No
Find the distance d of (x_c, y_c) of the circle from the line
Prog. 8
Fnctn 27
Select a pair of corners in the Rec (say P_1 P_2) defining a line
Not used when Prog. 19 is called by Prog. 21

Program 20 Checking for Possible Detection Opportunity between a Circular (C) Area Group (say No. m) and a Battle Group (BG) (say No. n)

Input: $(Lat, Long)_{i=1\ to\ u}$ for all u units in the battle group.

Lat_c, $Long_c$ - circular area group center coordinates.

R_c - circular area radius.

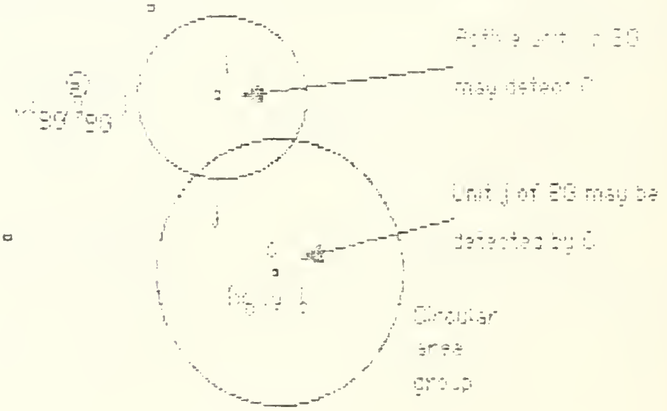
If any group active (radar and/or sonar) then need also:

δR_m, δS_m for extended ranges for the circular area group.

δR_{jn}, δS_{ln} for the extended ranges for the battle group, with an indication which of the u units is radar active (j) and which is sonar active (l).

Output: An indication of Det(m,n) and Det(n,m) to be equal 0 or 1 for detection opportunity designation. A Det(m,n)=1 flags indicate that group m has a detection opportunity against group n.

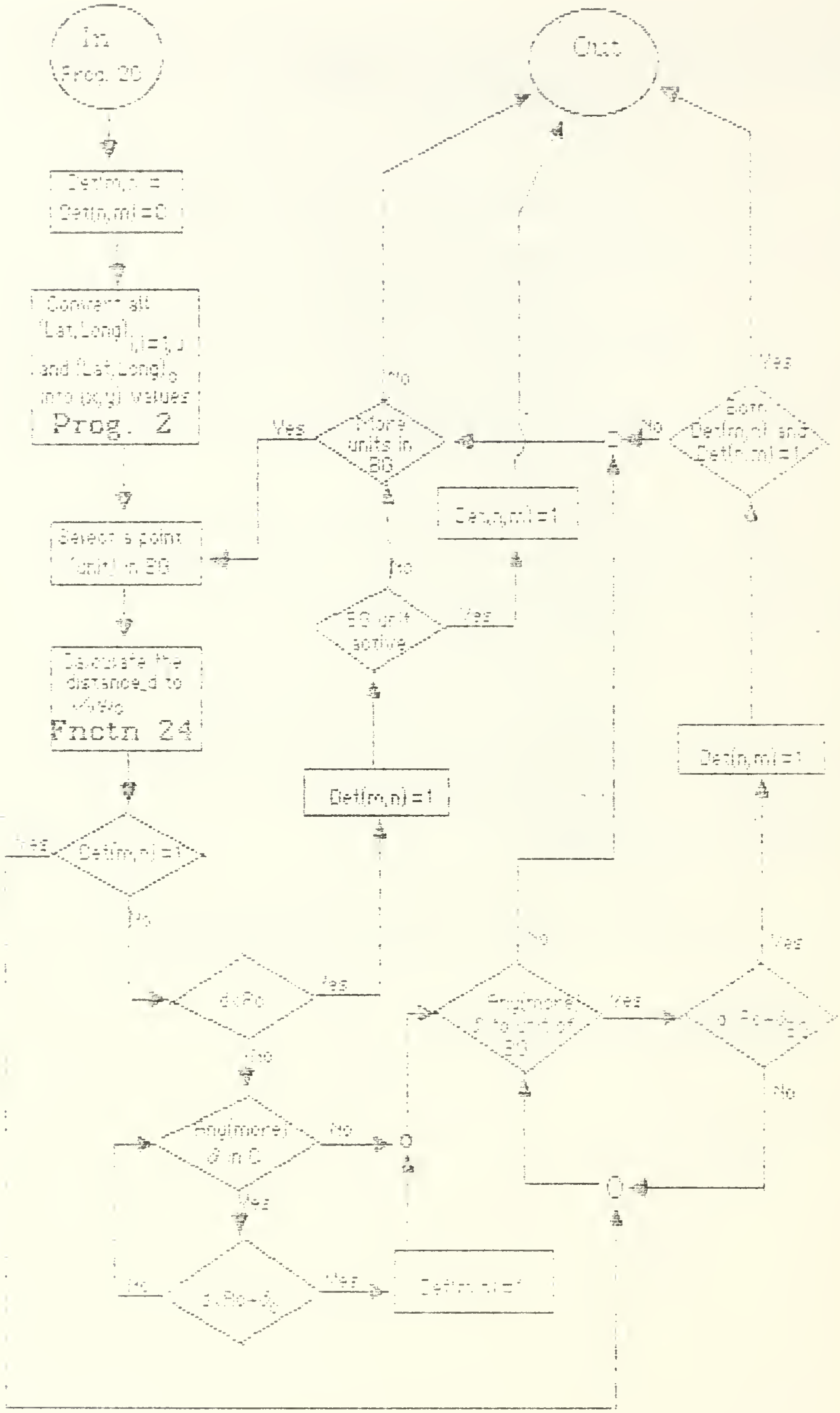
Procedure. Refer to the following diagram.



1. Select a [illegible] in the [illegible]
2. Calculate its distance from the [illegible] area (C) center (use Function 24)
3. If the unit is inside the said circle (distance < radius), set Det [illegible]=1
4. If both Det [illegible]=1 and the BB unit [illegible] else Det [illegible]=1 and go to step [illegible]
5. If Det [illegible]=1 check for the BB unit to be inside the extended range(s), if any, [illegible] determine again Det [illegible] if the unit is inside
6. Check if the BB unit is active, if not - go back to step 1.
7. Check the intersection between the [illegible] area and the BB's extended circle [illegible] SP [illegible] to determine Det [illegible] if intersection found.
8. Once both Det [illegible] are Det [illegible] there is no need for more BB unit analysis - go to step [illegible]
9. Go back to step 1 however, if there are no more units in the BB to be analyzed, go to step 10.
10. Finish analysis

Note: Better understanding of the process can be gained by following the flow diagram.

Diagram: See flow diagram


In
Prog. 2D
Det(m,n) =
Det(n,m) = 0
Convert all
(Lat,Long)$_i$, i = 1, J
and (Lat,Long)$_0$
into (x,y) values
Prog. 2
Select a point
(unit) in BG
Calculate the
distance d to
Fnctn 24
Yes
Det(m,n) = 1
No
d<Ro
Yes
No
Any(more)
j in C
No
Yes
No
Ro
Yes
Det(m,n) = 1
Det(m,n) = 1
BG unit
active
Yes
No
More
units in
BG
Yes
No
Det(n,m) = 1
Out
No
Any(more)
j to unit of
BG
Yes
Ro
Yes
No
Det(n,m) = 1
Both
Det(m,n) and
Det(n,m) = 1
Yes
No

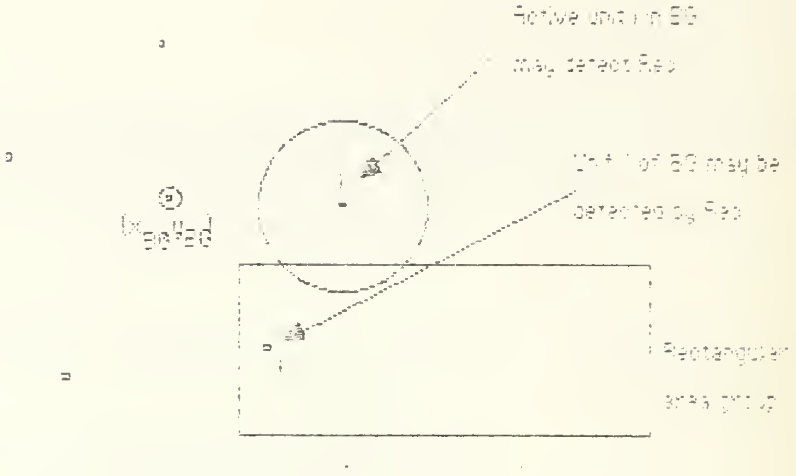
Program 21: Checking for Possible Detection Opportunity between a Rectangular S. Area Group [illegible] and a [illegible] Group [illegible]

Input
(Lat, Long)$_{k=1,K}$ for all units in the same group
(Lat, Long)$_{i=1,4}$ for the four corners of the rectangular area group
(LatR, LongR, LatS, LongS)$_{k=1,K}$ - the extended range rectangle corners (needed only for active radar or sonar)
R_m, S_m - the extended ranges of group [illegible] the barrier group [illegible] active unit [illegible] specified.

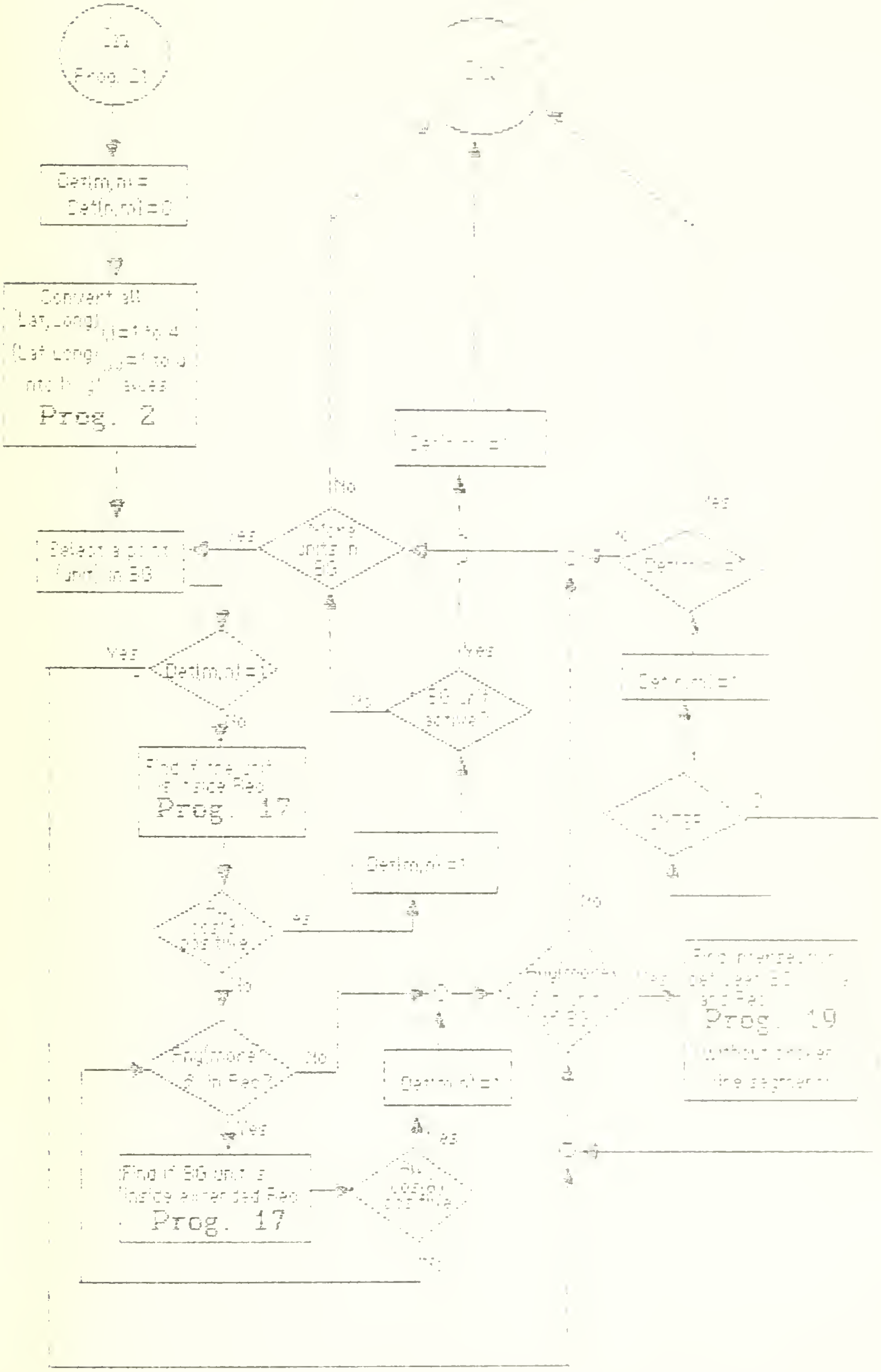
Output
An indication of Detect m,n and Detect n,m equal to 0 or 1 to designate detection opportunity. A Detect m,n flag indicates detection opportunity for group m against group n.

Procedure
The procedure is similar to that used in Program 20 except:
- In steps 2,3,5 check whether the [illegible] inside the rectangular area (Prog. 17) rather than a circular one
- In step 7 check the intersection between a rectangle and a circle (Prog. 19) instead of between two circles

Refer to the diagram on the next page.



Algorithm: See flow diagram.


In
Prog. 21
Detm,n) =
Det(n,m) = 0
Convert all
(Lat,Long)
Prog. 2
Select a point
unit in BG
Yes
No
units in
BG
Yes
Detm,n) = 1
No
Find if the unit
is inside Reg
Prog. 17
Detm,n) = 1
Any more
in Reg?
No
Yes
Find if BG unit is
inside extended Reg
Prog. 17
Yes
No
Yes
Detm,n) = 1
Detm,n) = 1
No
Find intersection
and Reg
Prog. 19
without dealing
line segment
Detm,n) = 1
Yes
No
Yes

Program 22 Checking for Possible Detection Opportunity between Two Battle Groups (say No. m and n)

Input: $(Lat_i, Long_i)_{i=1 \text{ to } u1}$ for all u_1 units in battle group No. m

$(Lat_j, Long_j)_{j=1 \text{ to } u2}$ for all u_2 units in battle group No. n

For all active units supply the extension range(s) (for radar and sonar in an "on" position), ∂R_{mi}, ∂S_{mi} and ∂R_{nj}, ∂S_{nj} for units in groups m and n respectively.

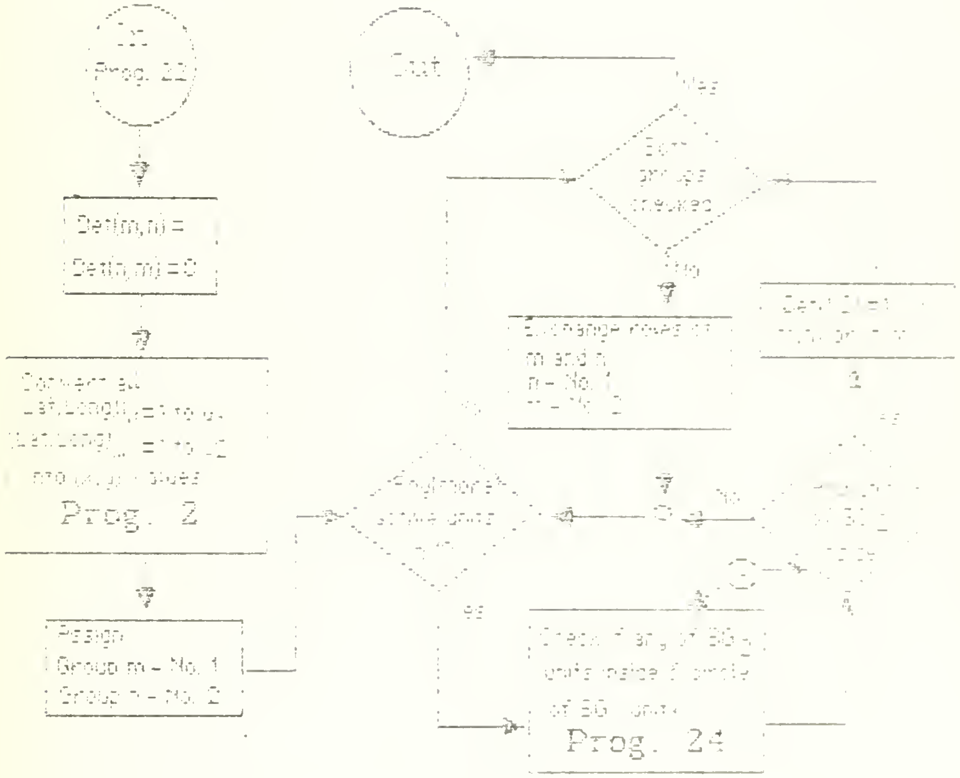
Output: An indication of Det(m,n) and Det(n,m) to be equal 0 or 1 to designate detection opportunity. A Det(m,n)=1 flag indicates the existence of detection opportunity for group m against group n.

Procedure:

1) Check one by one if any unit in one battle group is within the circular range(s) (radar and sonar) of the other battle group active units (using Function 24).
2) Exchange the roles of the battle groups and repeat step 1.

Note that a silent (passive - all radars and sonars "off") group cannot have a detection opportunity and need not be checked for one. Refer to the diagram on the next page.

Algorithm: See flow diagram



(*) There is no need to check all units of BG. Once a unit is found inside [illegible]

set Dent(1,2) = 1 and proceed

Program 23 Checking for Passive DF (Directional Measurement Opportunity (towards an Emitting Radar or Sonar) between Two Groups (say No. m and n)

Input: Position data for both groups.

$(Lat_i,Long_i)_{i=1\ to\ 4}$ for a battle group.

$(Lat_j,Long_j)_{j=1\ to\ 4}$ for a rectangular area group.

$(Lat_0,Long_0)$, R_0 for a circular area group.

δR_{mu}, δS_{mu} and δR_{nu}, δS_{nu} for the active groups or units.

R_p – passive detection range according to the case (drawn from the range tables).

Output: An indication of DF(m,n) and DF(n,m) to be equal 0 or 1 to designate directional measurement opportunity between groups m and n.

Procedure:

1) Select one of the two groups and check if active (radar or sonar "on"). If not, go to step 6.

2) Select the type of emission (radar or sonar) and the unit emitting (if in a battle group) as the center of the emittance. The emission location of an area group will be taken as its center x_0, y_0.

3) Establish R_p – the passive detection range according to the emission type (radar or sonar) and the DF equipment on the searcher group (from the appropriate range table).

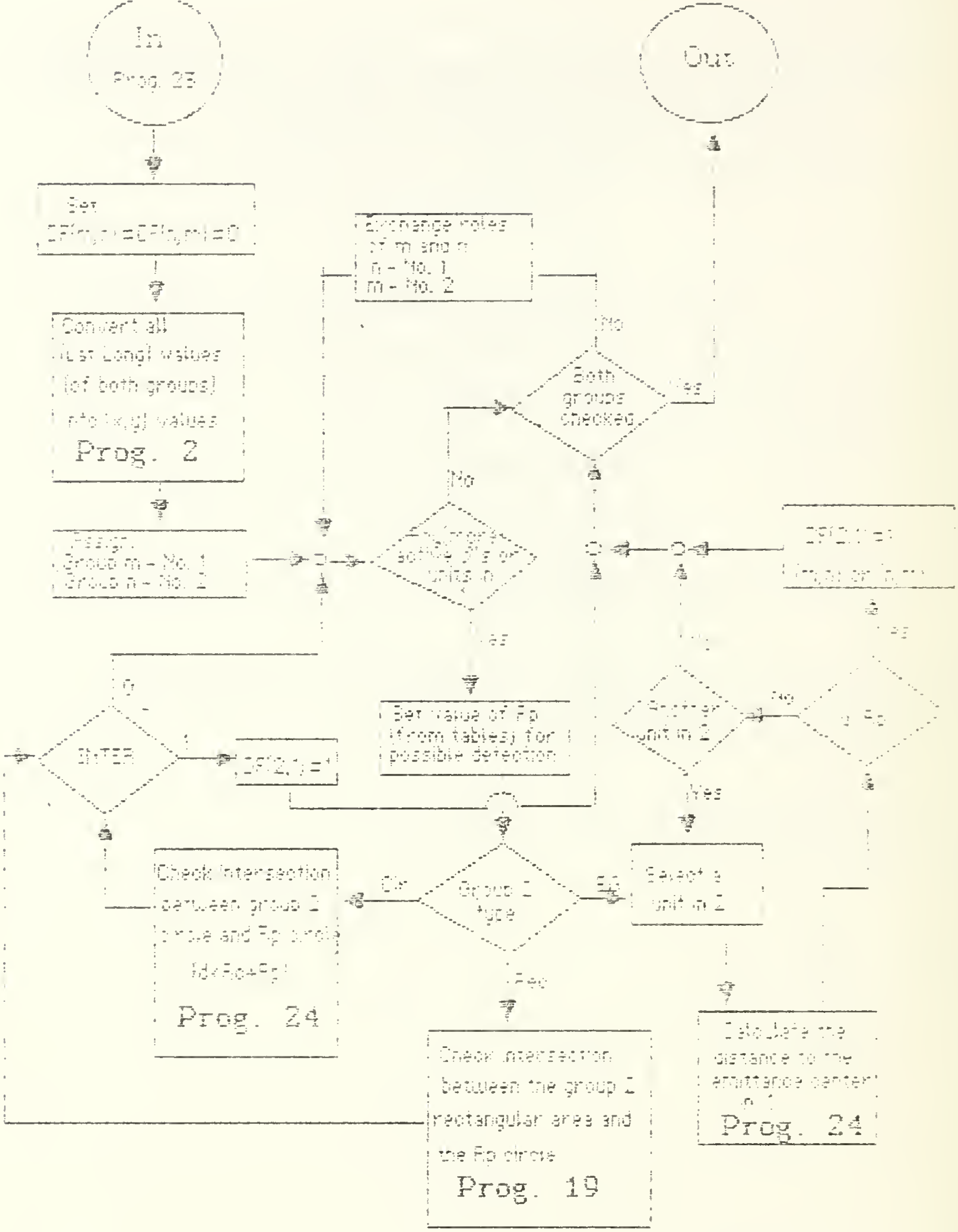
4) For each possible searching group, carry out the following according to its type:

- If a circular area searcher group, check the intersection between the circle of the area [illegible] and the F, [illegible] is [illegible] target group [illegible] (see Fig. [illegible] 24

- If a rectangular area is [illegible] the intersection between the [illegible] rectangle and the F, [illegible] the emitting target group or unit (see Fig. [illegible] 19)
- If a [illegible] of the 85 [illegible] the F, [illegible] emitting target group or unit (see Fig. 24

5. Repeat steps 3-4 for all other emitters/types; if there are no more proceed to step 6
6. Exchange the [illegible] have already been treated, proceed to step 7, else return to step 2
7) Finish the analysis [illegible] equal 0 or 1 according to the results found

Algorithm: See flow diagram.


In
Prog. 23
Set
DF(n,m)=DF(n,m)=0
Convert all
(Lat Long) values
(of both groups)
into (x,y) values
Prog. 2
Assign
Group m - No. 1
Group n - No. 2
Exchange roles
of m and n
n - No. 1
m - No. 2
No
Both
groups
checked
Yes
Out
No
Any more
active U's or
units in
Yes
Set value of Rp
(from tables) for
possible detection
0
INTER
1
DF(2,1)=1
Check intersection
between group 2
circle and Rp circle
(d<Ro+Rp)
Prog. 24
Cir
Group 2
type
Eq
Rec
Check intersection
between the group 2
rectangular area and
the Rp circle
Prog. 19
Select a
unit in 2
Calculate the
distance to the
emittance center
in 1
Prog. 24
Another
unit in 2
Yes
No
d < Rp
Yes
DF(2,1)=1
(m,n) or (n,m)

Function 24 [illegible]

[illegible]

Input [illegible]

Output The distance D

Function [illegible]

Function 25 [illegible]

Points 1 and 2 on a Sphere

Input LAT1, LONG1, LAT2, LONG2 [illegible]

Output The distance D [illegible]

Function D = 60 cos⁻¹ [illegible]

[illegible]

Notes 1) [illegible]

[illegible]

2) The factor [illegible]

[illegible]

3) [illegible]

[illegible]

4) [illegible]

the equation [illegible]

Function 26 Calculation to Find cos(θ) between Two Vectors $\bar{a}$ and $\bar{b}$

Input (x_a, y_a), (x_b, y_b) – Two vectors emerging from a common origin.

Output   cos(θ)

Function $$\cos(\theta) = (\bar{a}\cdot\bar{b}) / (|\bar{a}|\cdot|\bar{b}|) = \frac{x_a x_b + y_a y_b}{\sqrt{(x_a^2 + y_a^2)(x_b^2 + y_b^2)}}$$

Function 27 Calculation of the Distance d of a Point (x_p, y_p) from a Line Ax + By = C

Input (x_p, y_p) – the point coordinates
A,B,C – the line parameters.

Output   d – the distance from the point to the line.

Function $$d = (Ax_p - By_p - C) / (\pm\sqrt{A^2 + B^2})$$

(using ± according to the sign of C)

Function 28 Calculation of a Rhumb Line Course (Co) between Two Points

Input Lat_1, $Long_1$, Lat_2, $Long_2$ – the two points coordinates.

Output   Co – the course (°T)

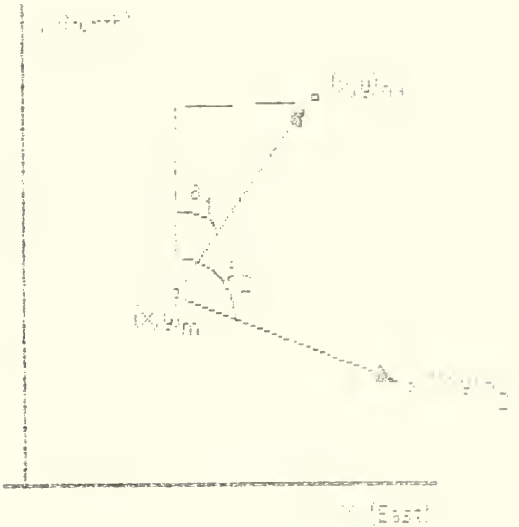
Function $DLo = Long_2 - Long_1$
$DL = Lat_2 - Lat_1$
$Co = \tan^{-1}((DLo \cdot \cos((Lat_1 + Lat_2)/2))/DL)$

Function 29 Direction Angle Calculation (Azimuth) Between Two Groups, say No. m and n

Input $(x_m,y_m),(x_n,y_n)$ – coordinates of the two groups (or units) – use centers for area groups

Output   The direction angle β from group m to group n (or vice versa if necessary), positive clockwise

Function: Refer to diagram



For the case of $y_n=y_m$

If $x_n>x_m$ then $\beta=90°$

If $x_n<x_m$ then $\beta=270°$ (or $-90°$)

For the case of $x_n=x_m$

If $y_n>y_m$ then $\beta=0$

If $y_n<y_m$ then $\beta=180°$

Otherwise $\beta = \tan^{-1}(x_n-x_m)/(y_n-y_m)$

4.3.2 Output Requirements.

We will use the following notation: D for disc storage, P for printout, and M for monitor screen display. M will be divided into M1 for point designation, M2 for directional lines, M3 - a circle, M4- a rectangle and M5 - a map display (to be incorporated later). In the following paragraphs we will list the types of outputs recommended in each of the boxes, as they appear in Figure 2.

Box C

- Game setup - Force structure, general initial position, equipment deployed (if relevent) - to be all stored on disc (D) for possible future replay of the game. A theatre map, if applicable, may also be stored at this point.
- All the above to be printed (P) for the umpire. The map can be displayed (M) with the general force positions.
- The printout (P) can be repeated on two separate pages relevant each to one of the players. They will be passed out to the players by the umpire.

Box G

- All state variable data (groups, configuration, sensor modes, tracks, plans, positions, speeds and courses) must be temporarily stored (D) by the umpire's instruction if he decides to exit into the engagement model. Upon completion of the engagement the kinematics/detection model can be restored for continuation of the game, with the required changes resulting from the engagement.

Box H

- The group arrangements and positions (Lat.Long), their speeds and courses as they result from the players instructions, are to be summarized in a printout (P).
- Each group area shape and/or configuration can be displayed (M1,M3,M4) upon request by the umpire. This should aid the

[illegible] will possibly output [illegible] the [illegible] [illegible] option exists.

- In the future a theatre map with all groups may be [illegible] [illegible].

Box 1

- Storage of all state variables on a disc (D) - time, position of groups, track and ENCOM plans - for game recording is needed possibly for post-game analysis, or for replay at a [illegible] decision point.

Box 2

- Same output as in Box 1 but one must add also the detection opportunity flags (0 and 1) for all relevant [illegible] [illegible] opportunity was identified (1). For automatic storage [illegible] data is suggested to be made every say 5 or [illegible] minutes unless some event such as detection, or player [illegible] changes in run or in decisions warrants it. For [illegible] the last two steps should be continuously retained on a temporary basis, preferably in memory if available, otherwise on a temporary disc (D). These will be continuously replaced by new data each time step.
- Printout (P) of all group positions and mode of operation [illegible] will be done automatically when a detection opportunity has just been identified, and will be [illegible] [illegible].
- A separate list of all pairs involved in detection opportunities [illegible] for the umpire's review.
- A graphical display (T) of pairs selected by the umpire with intersection or directional measurements to aid the umpire in judging whether the opportunity is realized.
- A separate printout (P) for each player summarizing the information he should obtain according to the umpire's [illegible] [illegible] [illegible] [illegible] as appropriate [illegible]

4.3.3 Graphics (M) Requirements.

Input Requirements (M_I)

- Only input of points (with the aid of the mouse) may be required (when this option will be included in the game). Note the three coordinate systems:

  (1) (X_M,Y_M) as appearing on the X-Y monitor screen display.

  (2) (X,Y) plane coordinates, extensively used in the mathematical algorithms.

  (3) (Lat,Long) geographical coordinate system used for navigational track plans.

  Conversion between systems 2 and 3 are made by programs 1 and 2. In general, it is suggested that the (Lat_0,$Long_0$) point of the group center will be used as the reference point in the conversion.

  Using coordinate system 1 requires a conversion into one of the other two systems (usually into No 2?). The conversion is a direct linear one left for the programming stage and is not discussed here.

Output Needs.

- Translating points from the model (X,Y) values into the monitor (X_M,Y_M) display coordinates.
- Drawing required: points, circles, rectangles and directional lines.

  Note that to draw an intersection between two area groups (say a circular and a rectangular one), there is no need for an absolute scaling but just for a relative one, possibly with a zooming option according to the umpire's desire. Printing inside the area a characteristic dimension of the group, and maybe also the group speed and course, may prove to be useful in aiding the umpire in his visual review.

Acknowledgement

This work was carried out with the support of the War Gaming Department in the Naval War College at Newport, RI.

Appreciation should be expressed to Professor J.N. Eagle of the Department of Operations Research at the Naval Postgraduate School for his useful suggestions throughout the whole model concept development.

References

Bowditch, Nathaniel, LL.D., "American Practical Navigator - An Epitome of Navigation", Vol. I, Chap. 7 - "The Sailings", Defense Mapping Agency Hydrographic Center (Pub. No. 9), 1977 Edition.

Couhat, Jean Labayle, Editor, "Combat Fleets of the World 1978/79 - Their Ships, Aircraft and Armament", Naval Institute Press, The United States Naval Institute, Annapolis, Maryland 1978.

Dunnigan, James F., "How to Make War - A Comprehensive Guide to Modern Warfare", William Morrow and Company Inc., New York 1982.

Kells, Lyman M. and Herman C. Stotz, "Analytical Geometry", Prentice-Hall, New York 1949.

DISTRIBUTION LIST

	NO. OF COPIES
Library (Code 0142) Naval Postgraduate School Monterey, CA 93943-5000	2
Defense Technical Information Center Cameron Station Alexandria, VA 22314	2
Office of Research Administration (Code 012) Naval Postgraduate School Monterey, CA 93943-5000	1
Center for Naval Analyses 2000 Beauregard Street Alexandria, VA 22311	1
Library (Code 55) Naval Postgraduate School Monterey, CA 93943-5000	1
Mr. H. A. Miller (Code 33T) Naval War College Wargaming Department Newport, RI 22314	1
LCDR H. R. Moore, USN (Code 3314B) Naval War College Wargaming Department Newport, RI 22314	3
Professor Haim Eldor (Code 55Eo) Naval Postgraduate School Monterey, CA 93943-5000	10
Operations Research Center, Rm E40-164 Massachusetts Institute of Technology Attn: R. C. Larson and J. F. Shapiro Cambridge, MA 02139	1

DUDLEY KNOX LIBRARY
3 2768 00302770 7